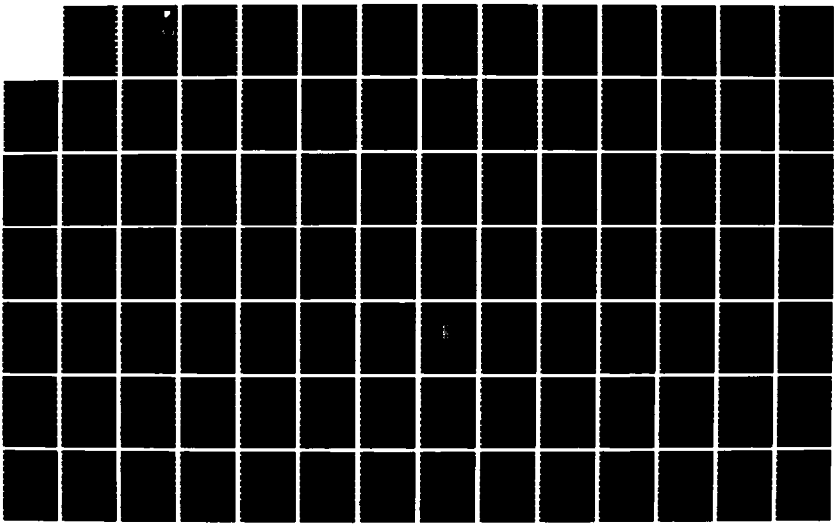
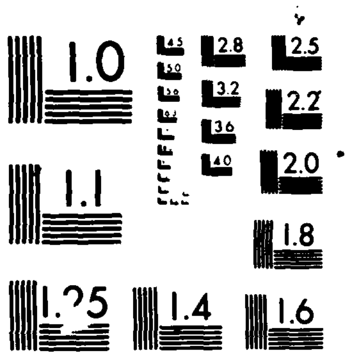
AD-A176 129 RESEARCH AND DEVELOPMENT FOR DIGITAL VOICE PROCESSING 1/2
(U) ARCON CORP WALTHAM MA J D TARDELLI ET AL. OCT 86
R86-01W RADC-TR-86-171 F19628-84-C-0024

UNCLASSIFIED F/G 17/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)

RADC-TR-86-171
Final Technical Report
October 1986

AD-A176 129

RESEARCH AND DEVELOPMENT FOR DIGITAL VOICE PROCESSING

ARCON Corporation

John D. Tardelli, C. M. Walter, J. T. Sims, P. A. LaFollette and P. D. Gatewood

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

This report has been reviewed by the RADC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

Although this report references * limited documents listed on pages 109 through 112, no limited information has been extracted.

RADC-TR-86-171 has been reviewed and is approved for publication.

APPROVED: Denis A Robitaille

DENIS A. ROBITAILLE, 1LT, USAF
Project Engineer

APPROVED:

ALLAN C. SCHELL
Chief, Electromagnetic Sciences Division

FOR THE COMMANDER: John A. Ritz

JOHN A. RITZ
Plans & Programs Division

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

A176129

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS N/A
2a. SECURITY CLASSIFICATION AUTHORITY N/A 2b. DECLASSIFICATION / DOWNGRADING SCHEDULE N/A	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
4. PERFORMING ORGANIZATION REPORT NUMBER(S) R86-01W	5. MONITORING ORGANIZATION REPORT NUMBER(S) RADC-TR-86-171
6a. NAME OF PERFORMING ORGANIZATION ARCON Corporation 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION Rome Air Development Center (EEV)
6c. ADDRESS *(City, State, and ZIP Code)* 260 Bear Hill Road Waltham MA 02154	7b. ADDRESS *(City, State, and ZIP Code)* Hanscom AFB MA 01731-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Rome Air Development Center 8b. OFFICE SYMBOL *(If applicable)* EEV	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F19628-84-C-0024
8c. ADDRESS *(City, State, and ZIP Code)* Hanscom AFB MA 01731-5000	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. 33401F PROJECT NO. 7820 TASK NO. 03 WORK UNIT ACCESSION NO. 50

11. TITLE *(Include Security Classification)*
RESEARCH AND DEVELOPMENT FOR DIGITAL VOICE PROCESSING

12. PERSONAL AUTHOR(S)
J. D. Tardelli, C. M. Walter, J. T. Simms, P. A. LaFollette, P. D. Gatewood

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
Final	FROM Jan 84 TO Feb 86	October 1986	120

16. SUPPLEMENTARY NOTATION
N/A

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP
09	04	
12	02	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Digital Voice Processing; Signal Processing; Speech Compression; Canonical Coordinate Analysis; Vocoder (See Reverse)

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

This technical report covers a variety of topics and development areas. Original research on the decomposition of a Canonical Coordinate Transformation process based on non-Euclidean error minimization criteria is covered. The implementation of an algorithm resulting from this work and its application to sample error metrics is described. The definition of an algorithm resulting from this work and its application to sample error metrics is described. The definition of a spectral moment error metric and the development of statistical and empirical non-Euclidean error criteria are covered. Test results from several experimental vocoders using this algorithm and specific error metrics are presented. The results of original research on the characterization of the acoustic background noise for several Air Force platforms is covered in depth. A study of Linear Predictive Coding improvements and their implementations at RADC (EEV) is given along with details on the installation of several array processor vocoder algorithms. The status and latest developments of processor hardware, operating systems and program development software are documented. New software

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Denis A. Robitaille, 1Lt, USAF	22b. TELEPHONE *(Include Area Code)* (617) 377-2068 22c. OFFICE SYMBOL RADC (EEV)

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

tools that provide system control of the Spectral Dynamics SD350 Signal Processor, the Adams-Russell Speech Processing Peripheral and Precision Filter Set are described. The introduction and improvements to the Interactive Laboratory System (ILS) analysis and display package are covered along with its relationship to the current Speech Data Base library used at RADC (EEV). New communicability and vocoder audio/control/data systems at the Speech Processing facility are described.

18. SUBJECT TERMS. (Continued)

Principal Component Analysis	Improved Linear Predictive Coding
Non-Euclidean Error Minimization	Spectral Analysis
Error Metric	Speech Data Base
Aircraft Acoustic Noise	Communicability Testing System
Noise Suppression	Speech Processing Peripheral
Speech Processing Facility	ILS

17. COSATI CODES (Continued)

Field	Group
17	02

CONTENTS

CONTENTS (Continued)

CHAPTER 1

OVERVIEW

1.1 INTRODUCTION

This is the Final Technical Report describing work performed for the U.S. Air Force Systems Command RADC/EEV Speech Processing Facility at Hanscom AFB, MA. This work was performed under contract number F19628-84-C-0024 during the period 1-January-1984 through 14-February-1986. This effort is a continuation and extension of work performed under previous contracts with RADC/EEV and reported elsewhere (Ref. 1.1-1.9).

This technical report will cover a variety of topics and development areas. Original research on the decomposition of a Canonical Coordinate Transformation process based on non-Euclidean error minimization criteria will be covered. The implementation of an algorithm resulting from this work and its application to sample error metrics will be described. The definition of a spectral moment error metric and the development of statistical and empirical non-Euclidean error criteria will be covered. Test results from several experimental vocoders using this algorithm and specific error metrics will be presented. The results of original research on the characterization of the acoustic background noise for several Air Force platforms is covered in depth. A study of Linear Predictive Coding improvements and their implementations at RADC/EEV is given along with details on the installation of several array processor vocoder algorithms. The status and latest developments of processor hardware, operating systems and program development software will be documented. New software tools that provide system control of the Spectral Dynamics SD350 Signal Processor, the Adams-Russell Speech Processing Peripheral and a Precision Filter Set will be described. The introduction and improvements to the Interactive Laboratory System (ILS) analysis and display package will be covered along with its relationship to the current Speech Data Base library used at RADC/EEV. New communicability and vocoder audio/control/data systems at the Speech Processing facility will be described.

Numerous software packages are referred to within this report. Source code for all programs developed under this contract is available to authorized users of the RADC/EEV Speech Processing Facility through reference to the virtual disk REPT86 and directory file [200,200]REPT86.DIR.

1.2 OTHER CONTRACT TASKS

During the course of this contract numerous requirements were met that do not lend themselves to a descriptive section in this technical report. Several of these work areas are covered in the following paragraphs for completeness.

Voice Processor Intelligibility Testing - ARCON provided the training, staff and supervision for the RADC/EEV in-house voice communication systems test and evaluation program. This program utilizes the Diagnostic Rhyme Test (DRT) as a measure of system intelligibility. Twice weekly DRT listener sessions were conducted and the data was collected, scored, analyzed, reported on and stored in the DRT Data Base. All software and data for this system was maintained throughout the period of this contract. Evaluation of test data for both in-house research staff and other government users of the RADC/EEV facility was also provided. Numerous in-house DRT tapes were prepared for system evaluation at both the EEV facility and an independent contractor (DYNASTAT Corp., Austin, TX). During the period of this contract an extensive study of the effects of equalizing the speaker presentation levels to DRT listeners and voice processing systems was performed by ARCON ›nnel (Ref. 1.10). This study and modifications to scoring procedures, data base structure, associated softw ›xtended analysis methods are fully covered in Ref. 1.11.

System Performance Evaluation - This task led to ARCON's involvement in the design and planning of a series of tests on the ANDVT digital voice processor in Air Force operational environments. A March 1984 ARCON memo to C.P. Smith of RADC/EEV had detailed some ideas for an In-Place DRT Procedure using the TRS80 M100 Data Entry Units. The goal was to have Speakers and Listeners in various environments, both on the ground and in the air, communicating over actual Air Force channels conduct a modified version of the DRT. This test method and procedure was suggested for use during the ESD ANDVT evaluation and accepted. Over the following four month period ARCON provided support to the project that included the following:

1. Complete M100 Software Development
2. Development of data transfer and analysis software
3. Assistance with experimental design
4. Definition of the DRT Field Equipment Set
5. Training of ESD and RADC speaker/listeners
6. Generation of field training procedures for operational speaker/listeners
7. Development of a single-speaker DRT Data Base for the results
8. Development of Procedures for processing results
9. Processing of Field data
10. Verification analysis of the In-Field DRT

The field data that was received was incomplete because of equipment malfunctions ranging from software bugs to aircraft that could not fly. Program schedules were unrealistic and baseline data unattainable from some listener/speakers because of their other commitments. All of these problems resulted in the inability to verify or even relate the field test results to in-house tests made on audio recordings returned from the field. The basic experimental design and DRT Field Equipment Set as specified for this effort were not at fault in the inability to verify the relationship between in-field and in-house DRTs. ARCON did not have the responsibility for the reporting of this effort. The positive fallout of this work was: i) several of the acoustic noise recordings and measurements utilized in Chapter 3 of this report ii) the In-Field DRT software and procedures reported in Ref. 1.11.

DoD Digital Voice Processing Consortium Support - Work in this area focused on the preparation of DRT and DAM (Diagnostic Acceptability Measure) test material for the evaluation of two 16 kbps vocoder systems and the ADPCM digital switch. This work required travel to Virginia and New Jersey. One of the vocoders was tested by modifying the RADC/EEV's MAP-300 array processor to run the algorithm software.

Other work in this area consisted of the preparation of eight DAM test series for submittal to DYNASTAT and the digital recording of new DAM test sentences at DYNASTAT. Technical assistance was provided in the development of a library of DRT and DAM digital master tapes. This process included the equalization of speaker presentation levels on the digital tapes using a method developed by Mr. James Sims of ARCON (Refs. 1.9 and 1.12). The resulting library will be detailed in a future RADC report.

1.3 ACKNOWLEDGMENTS

One of the authors, Mr. Charlton Walter, worked as a private consultant to ARCON Corporation while performing research on Canonical Coordinate Based Data Compression Methods. This work is detailed in Chapter 2 of this report.

The authors wish to acknowledge the cooperation, encouragement and technical support received from Mr. Anton Segota and 1st Lt. Denis Robitaille of RADC/EEV. The electronics for the Communicability Test Facility detailed in Chapter 7 were designed and constructed by Lt. Robitaille.

CHAPTER 2

CANONICAL COORDINATE BASED DATA COMPRESSION

For the past six years ARCON Corporation has been performing original research for RADC/EEV on Time/Frequency Domain Interrelationships with the ultimate goal to provide a digital speech compression algorithm that can utilize non-Euclidean error minimization criteria and be formulated in a parallel or matrix manner to make use of a new generation of true parallel signal processors. These processors are just now beginning to make their mark in the signal processing field. The ARCON algorithm research covered many areas before isolating a particular decomposition of the canonical coordinate (CC) domain (Refs. 2.1 - 2.4). The CC domain is defined by that space in which the error metric or criteria and the signal correlation matrix are diagonal. This decomposition takes advantage of a pseudo-canonical coordinate parameter and an ordering procedure to meet the needs of an analysis - synthesis bandwidth compression communications system. The algorithm has been successfully implemented on the RADC/EEV FPS AP120-B array processor in the form of the program CCVOC. The program will accept any Hermitian error metric for definition of the error criteria, transform input speech files into the defined CC domain and synthesize output speech files at compression ratios set by the operator. Initial work has been done using the identity matrix I as the error metric. This reduces the CC analysis to the Euclidean Principal Component analysis. The definition of a simple non-Euclidean error metric G based on the long term statistical spectral moment of speech is given in this report. A procedure for the generation of the G error metric has been implemented on the AP120-B. Empirical error metrics based on the psychoacoustic affects utilized in Channel Vocoder designs have been developed. All of these metrics have been used with the program CCVOC to process speech. The resulting speech has been evaluated for intelligibility.

2.1 A CANONICAL COORDINATE BASED VOCODER ALGORITHM

It has been known for some time that simple Euclidean mean-squared error minimization criteria were not the most optimum criteria to use for the design of speech analysis and reconstruction systems, but were easily implemented using recursive procedures running on primarily sequential processors. However, the advent of systolic array processor technology has opened the possibility of applying matrix array techniques to real-time waveform analysis and synthesis utilizing non-Euclidean error minimization procedures that can be tailored to specific speech compression problems.

2.1.1 Some Basic Signal Vector Transformation Interrelationships in the Time, Frequency and Canonical Coordinate Domains

The basic signal analysis and reconstruction process to be utilized in this paper is one that treats the original sampled waveform data at a segmented series of consecutive N+1 dimensional data vectors $x(t) = (x_o(t), \ldots, x_N(t))^t$, indexed on a frame number t, and whose complex valued components, $x_n(t)$, are generally correlated and will be required to satisfy a non-euclidean, quadratic error metric, characterized by an hermitian matrix J in the time domain.

These vectors are first transformed into a frequency domain representation y, through a Fourier transform process Z, as

$$y(t) = Z\, x(t) \quad , \tag{1}$$

where Z is a unitary DFT maxtrix having components

$$Z_{mn} = (1/N) \exp(-2\pi i\; mn/N) \; .$$

The vectors are then transformed into a succession of special pseudo-canonical and *canonical coordinate* domains, characterized by the vector $\beta(t)$, $\hat{\beta}(t)$ and $z(t)$, that simplifies the reconstructed signal error minimization process, by achieving a more compact representation of the signal x, or its DFT y, in the β, $\hat{\beta}$ and z domains.

Let R be the correlation matrix

$$R = \langle x(t)\; x^{\dagger}(t) \rangle \quad , \tag{2}$$

defined over an ensemble of vectors, $\{x(t)\}$, indexed on the frame number t, where $\langle\cdot\rangle$ represents an averaging process over the ensemble. Let S be the "correlation" matrix in the y-domain

$$S = \langle y(t)\; y^{\dagger}(t) \rangle \; . \tag{3}$$

Then from (1) we have that $\langle yy^{\dagger} \rangle = Z \langle xx^{\dagger} \rangle Z^{\dagger}$, or

$$S = ZRZ^{\dagger} \; . \tag{4}$$

Here S is often referred to as a spectral matrix. Note, in particular, that if R is circularly symmetric, then Z diagonalizes R, hence S will be diagonal in this situation.

Now in terms of the above ensemble averaging process, let $x(t)$ satisfy a quadratic error criterion, in the x-domain, of the form

$$E_{\hat{x}} = \langle \Delta x^{\dagger} J \Delta x \rangle = \langle (x(t) - \hat{x}(t))^{\dagger} J (x(t) - \hat{x}(t)) \rangle \,, \tag{5}$$

where $\hat{x}(t)$ is an estimate of $x(t)$, and J is the hermitian error metric in the x-domain. This error can be expressed in terms of an error metric K in the y-domain as

$$E_{\hat{y}} = \langle \Delta y^{\dagger} K \Delta y \rangle = \langle (y(t) - \hat{y}(t))^{\dagger} K (y(t) - \hat{y}(t)) \rangle \,, \tag{6}$$

where $\hat{y}(t) = Z\,\hat{x}(t)$ and K will be related to J by the equation

$$K = ZJZ^{\dagger} \,. \tag{7}$$

If we now let V be a unitary matrix that transforms K to the diagonal form Γ by means of the Eigenvector equation

$$V^{\dagger} K V = \Gamma \quad \text{or} \quad KV = V\Gamma \,, \tag{8}$$

and then let $\beta(t)$ be a vector such that

$$y(t) = V\beta(t) \quad \text{or} \quad \beta(t) = V^{\dagger} y(t) \,. \tag{9}$$

Then in terms of the pseudo-canonical coordinate domain β, the quadratic error $E_{\hat{y}}$ in the y-domain, with metric K, will have the form

$$E_{\hat{\beta}} = \langle \Delta\beta^{\dagger} \Gamma \Delta\beta \rangle = \langle (\beta(t) - \hat{\beta}(t))^{\dagger} \Gamma (\beta(t) - \hat{\beta}(t)) \rangle \,, \tag{10}$$

where $\Gamma (= V^{\dagger} K V)$ is a diagonal error metric.

Let T be the correlation matrix in the β-domain

$$T = \langle \beta(t)\, \beta^{\dagger}(t) \rangle \,, \tag{11}$$

then from (9) we have

$$T = V^{\dagger} S V \,. \tag{12}$$

Note that T will generally not be diagonal, hence β will not be a full canonical coordinate, since this requires that both T and Γ be diagonal. However, the diagonality of Γ does provide us with a mechanism for ordering the components of the estimate $\hat{\beta}(t)$ in such a way as to minimize the [illegible] errors E_x or E_y, where $x(t) = Z^{-1} y(t) = Z^{-1} V \beta(t)$.

For certain applications, two true canonical coordinate vectors, $\delta(t)$ and $z(t)$ need to be defined. Let $\delta(t)$ be such that

$$\beta(t) = H\,\delta(t) \qquad (13)$$

where H is an hermitian matrix and where H is chosen so that $\delta(t)$ will have a "flat" diagonal correlation matrix, or "flat spectrum" in the δ-domain, i.e.,

$$\langle\delta(t)\;\delta^{\dagger}(t)\rangle = I\;. \qquad (14)$$

Then from (11), (12) and (14) and the condition that H is hermitian (i.e., $H^{\dagger}=H$) we see that H^2 satisfies the relationships

$$T = \langle\beta(t)\,\beta^{\dagger}(t)\rangle = H\,\langle\delta(t)\;\delta^{\dagger}(t)\rangle\,H^{\dagger} = HH^{\dagger} = H^2\;, \qquad (15)$$

and hence from (12)

$$T = V^{\dagger}SV = H^2\;. \qquad (16)$$

Now let $z(t)$ be such that

$$z(t) = \Lambda^{\frac{1}{2}}\;\delta(t)\;, \qquad (17)$$

where Λ is a real diagonal matrix, and where z also satisfies a simple euclidean error metric of the form

$$E_{\hat{z}} = \langle\Delta z^{\dagger}\Delta z\rangle = \langle(z(t) - \hat{z}(t))^{\dagger}\;(z(t) - \hat{z}(t))\rangle\;. \qquad (18)$$

Then from (18) we see that δ satisfies a weighted euclidean metric of the form

$$E_{\hat{\delta}} = \langle\Delta\delta^{\dagger}\Lambda\Delta\delta\rangle\;, \qquad (19)$$

where Λ is the diagonal error metric in the δ-domain. From (14) the z-domain will have a diagonal correlation matrix of the form

$$\langle z(t)\;z^{\dagger}(t)\rangle = \Lambda\;. \qquad (20)$$

Finally, since

$$\langle \Delta\beta^{\dagger}\Gamma\Delta\beta \rangle = \langle \Delta\delta^{\dagger}H^{\dagger}\Gamma H\Delta\delta \rangle = \langle \Delta z^{\dagger}\Lambda^{-\frac{1}{2}}H^{\dagger}\Gamma H\Lambda^{-\frac{1}{2}}\Delta z \rangle$$

$$= \langle \Delta z^{\dagger}\Delta z \rangle$$

it follows that $\Lambda^{-\frac{1}{2}} H^{\dagger}\Gamma H\Lambda^{-\frac{1}{2}} = I$ or

$$H^{\dagger}\Gamma H = \Lambda \tag{21}$$

These coordinate domain relationships are shown in a simplified diagramatic form in Figure 2.1. Note that this approach has several significant differences from that used in our 1982 ICASSP paper (Ref. 2.1) in order to simplify the computational process.

2.1.2 Signal Analysis and Reconstruction Using Pseudo-Canonical Coordinates

From the previous section we see that the pseudo-canonical coordinate β provides the first signal representation domain in which the error metric Γ is guaranteed to be in diagonal form, and hence provide the following procedure for ordering the components of the estimator $\hat{\beta}(t)$ in such a way as to minimize the non-euclidean error $E_{\hat{x}}$, or $E_{\hat{y}}$, where $\hat{x}(t) = Z^{+}\hat{y}(t) = Z^{+}V\,\hat{\beta}(t)$.

In particular, let $\hat{\beta}(t,p)$ be an estimated vector consisting of the first p components of $\beta(t)$, followed by N-p+1 time invariant estimates for the remaining components that are not transmitted for use in the resynthesis process, i.e.,

$$\hat{\beta}(t,p) = \text{col.}\ (\beta_o(t), \ldots, \beta_{p-1}(t),\ \beta_p, \ldots,\ \hat{\beta}_N)\ . \tag{22}$$

Then

$$E_{\hat{\beta}}(p) = \langle \Delta\beta^{+}\Gamma\Delta\beta \rangle = \sum_{m,n=0}^{N} \langle \Delta\beta_m^{*}(t,p)\,(\gamma_m\delta_{mn})\ \Delta\beta_n(t,p) \rangle$$

$$= \sum_{n=p}^{N} \gamma_n \langle (\beta_n(t)-\hat{\beta}_n)\quad (\beta_n(t)-\hat{\beta}_n)^{*}$$

$$= \sum_{n=p}^{N} \gamma_n (\quad \beta_n(t)\beta_n^{*}(t) \quad - \quad \beta_n(t) \quad \beta_n^{*}(t) \quad) \tag{23}$$

Inner Product Scalar Constraint Relations	Vector Variable Name		Transformation Matrix Name	Outer Product Correlation Matrix Relations
$x^\dagger Jx=1$	Raw Signal Vector	x		$\langle xx^\dagger\rangle=R$
		$\uparrow$ $\boxed{Z^\dagger}$ $\uparrow$	Inverse DFT ($ZRZ^\dagger=S$)	
$y^\dagger Ky=1$ ($K=ZJZ^\dagger$)	DFT Signal Vector	y		$\langle yy^\dagger\rangle=S$
		$\uparrow$ $\boxed{V}$ $\uparrow$	Metric Diagonalization Transformation ($V^\dagger KV=\Gamma$)	
$\beta^\dagger\Gamma\beta=1$	Pseudo Canonical Coord. Vector	β		$\langle\beta\beta^\dagger\rangle=T$
		$\uparrow$ $\boxed{H}$ $\uparrow$	($V^\dagger SV=T=H^2$) Pseudo Canonical Correlation Transformation ($H=V^\dagger S^{\frac{1}{2}}V$)	
$\delta^\dagger\Lambda\delta=1$	Spectrally Flat Canonical Coord. Vector	δ		$\langle\delta\delta^\dagger\rangle=I$
		$\uparrow$ $\boxed{\Lambda^{-\frac{1}{2}}}$ $\uparrow$	Canonical Coord. Eigenspectrum Normalization Transformation ($\Lambda=H^\dagger\Gamma H$)	
$z^\dagger z=1$	Orthogonalized Canonical Coord. Vector	z		$\langle zz^\dagger\rangle=\Lambda$

Error Metric and Resynthesis Error Relations in the x,y,β,δ and z Domain

$$x^\dagger Jx = y^\dagger Ky = \beta^\dagger\Gamma\beta = \delta^\dagger\Lambda\delta = z^\dagger z$$

$$E=\langle\Delta x^\dagger J\Delta x\rangle=\langle\Delta y^\dagger K\Delta y\rangle=\langle\Delta\beta^\dagger\Gamma\Delta\beta\rangle=\langle\Delta\delta^\dagger\Lambda\Delta\delta\rangle=\langle\Delta z^\dagger\Delta z\rangle$$

Figure 2.1

Interrelationship Between the Time, Frequency and

Canonical Coordinate Domains

and if we normalize $\beta(t)$ so that $\langle\beta(t)\rangle = 0$, i.e.,

$$\langle\beta_n(t)\rangle=0 \text{ for all } n=0,\ldots, N \quad , \tag{24}$$

then the above expression reduces to

$$E_{\hat{\beta}}(p) = \sum_{n=p}^{N} \gamma_n H_{nn}^2 = \sum_{n=p}^{N} \mu_n \tag{25}$$

where, from (15)

$$\langle\beta_n(t) \;\; \beta_n^*(t)\rangle = T_{nn} = H_{nn}^2 \tag{26}$$

and where the component significance measure μ_n is

$$\mu_n = \gamma_n H_{nn}^2 \tag{27}$$

Thus $E_{\hat{\beta}}$ (p) will be minimized for each p, providing the coefficients μ_n are ordered so that

$$\mu_o \geq \mu \geq \ldots \geq \mu_N \quad . \tag{28}$$

From (16) we have a convenient procedure for calculating H_{nn}^2 as

$$H_{nn}^2 = v^+(n) \;\; Sv(n) \tag{29}$$

where v(n) is the n-th Eigen vector of (8), i.e., $Kv(n) = \gamma_n v(n)$. Moreover, if S is diagonal, then

$$H_{nn}^2 = s_n(v^\dagger(n)v(n)) \quad . \tag{30}$$

In the specific implementation of the above analysis and reconstruction process, we have two major alternative courses of action. One course is based on the assumption that long term averaging of DFT vectors in the y-domain is meaningful. This leads to relatively time independent values for $H^2 = T = \langle\beta(t)\beta^+(t)\rangle$ in the β-domain that can be used, in conjunction with a time independent Γ to define a set of time independent β-component significance measures $\mu_n = \gamma_n H_{nn}^2$. This provides an explicit, time independent procedure for ordering the coordinates in the pseudo-canonical coordinate representation in such a way as to minimize the non-euclidean error for any given number of transmitted components of β. Since the transformation matrix V will be available at both analysis and reconstruction sites, only β-component values, in a specified predetermined order need be transmitted.

However, in the event that long term spectral moment averaging over framed data in the y-domain is only meaningful to a first approximation, as in the case of speech and other signals having significant information imbedded in the short term time varying structure, another course of action in implementing the above analysis and resynthesis process is possible. In particular, for each frame of data $x(t)$, construct the DFT $y(t) = Zx(t)$ using a "zero fill" process on the x-data. Then, on the assumption that $R(t) = \langle x(t)x^{\dagger}(t)\rangle$ is defined by "averaging" over one circularly shifted frame of data, we have that $R(t)$ is circularly symmetric and hence $S(t) = Z\,R(t)Z^{\dagger}$ is diagonal. Consequently, from (27) and (30) we have that the component significance measure for each frame t can be expressed as

$$\mu_n(t) = \gamma_n(v^{\dagger}(n)v(n)s_n(t)$$

where $s_n(t) = |y_n(t)|^2$ is the n-th power spectral component associated with frame t. Then for each frame t, the $\mu_n(t)$ are ordered as in (28), by a frame dependent permutation vector $P(t)$. The first p of the reordered values of $\beta(t)$ are transmitted, together with the first p values of the permutation vector and a pair of frame scaling values, as required. These parameters are then used to reconstruct a p-th order approximation $\hat{\beta}(t,p)$, then $\hat{y}(t,p)$ and finally $\hat{x}(t,p)$.

2.1.3 Principal Component Analysis as a Special Case of Pseudo-Canonical Coordinate Analysis for Euclidean Error Metric

If we let J=I, corresponding to an euclidean metric in the x=domain, then from (7) we see that K is also an euclidean metric in the y-domain since

$$K = ZJZ^{+} = ZZ^{+} = I. \tag{31}$$

The Eigenvector procedure $V^{+}KV = \Gamma$ for determining a unitary matrix V and diagonal matrix Γ then reduces to the form yielding an identity matrix for Γ, i.e.,

$$\Gamma = V^{+}KV = V^{+}V = I\ , \tag{32}$$

and leaves V indeterminant at this point.

To remove this indeterminacy in V we note from (21) that the diagonal matrix Λ takes on the form

$$\Lambda = H^{\dagger}\Gamma H = H^{\dagger}H = H^2 \;, \tag{33}$$

hence H^2 is diagonal, and from (16) it follows that

$$V^{\dagger}SV = H^2 = \Lambda \;. \tag{34}$$

Therefore V is a matrix of Eigenvectors of S and the diagonal elements of Λ are the associated Eigenvalues.

Putting this in the more conventional time domain context we can write, substituting $S = ZRZ^{\dagger}$ into (34), that $V^{\dagger}ZRZ^{\dagger}V = \Lambda$, and letting

$$U = Z^{\dagger}V \;, \tag{35}$$

yields the conventional principal component Eigenvector equation

$$U^{\dagger}RU = \Lambda \;, \tag{36}$$

defining a transformation

$$x(t) = U\beta(t) \tag{37}$$

between the time domain x and the principal component domain β with component significance weighted on the size of the diagonal elements of Λ.

The resynthesis error, using only p of N components will then be, from (25)

$$E_{\hat{\beta}}(p) = \sum_{n=p}^{N} u_n = \sum_{n=p}^{N} H^2_{nn} = \sum_{n=p}^{N} \lambda_n \tag{38}$$

For the important case where R is circularly symmetric, it can be shown that $U=Z^{\dagger}$ will diagonalize R, yielding

$$\Lambda = U^{\dagger}RU = ZRZ^{\dagger} = S \;, \tag{39}$$

hence

$$V = ZU = ZZ^{\dagger} = 1 \;, \tag{40}$$

and since

$$y(t) = V\beta(t) = \beta(t) \;, \tag{41}$$

it follows that the principal component domain β is the DFT of y-domain, with components weighted on the power spectrum values $\mu_n = \lambda_n = s_n$ for n=0,..., N.

Note also that in the principal component domain (i.e., J=K=I) we have the same options discussed in Section 3 of dealing with a correlation matrix R that is based on a long term signal vector averaging process, as in the typical Loevé Karhunen analysis, or of using a circular averaging process on single frames of data x(t), in the x-domain, leading to a circularly symmetric R(t) and hence to a diagonal spectral matrix S(t) whose diagonal elements are the power spectrum components $s_n(t) = |y_n(t)|^2 = |\beta_n(t)|^2$, with $\mu_n(t) = s_n(t)$.

2.2 A BASIC NON-EUCLIDEAN ERROR METRIC UTILIZING LONG TERM AVERAGE SPECTRAL MOMENTS

While some broad general psychoacoustic attributes of the speech signal appear to be definable in the frequency domain, it appears that these attributes have a very large number of interrelated parameters associated with them. And since we also have an exceedingly large number of degrees of freedom available in the specification of an error metric in the frequency domain, it appears that highly systematic procedures are needed to converge on the determination of a desirable metric. Otherwise, we are faced with the classic problem of almost endless "fiddling" and suboptimization of parametrs that has beset the narrowband voice area for a long time.

One systematic possibility, which this note begins to address, is based on the use of statistical considerations to define an initial error metric, based on purely statistical error minimization techniques, and then to provide a systematic variational mechanism of perturbing this metric while listening to the resynthesized voice signal.

2.2.1 Error Characterization in Terms of Error Metric K and Cross Spectral Moment Matrix $\tilde{G}$ in the Frequency Domain y

In the frequency domain the resynthesis error $E_{\hat{y}}$ is defined in terms of the error metric K and a resynthesis estimation process $\hat{y}$ as the quadratic norm

$$E_{\hat{y}} = 1/2\langle \Delta y^{+} K \Delta y \rangle = 1/2\langle (y-\hat{y})^{+} K (y-\hat{y}) \rangle$$

$$= 1/2(\langle y^{+}Ky\rangle + \langle \hat{y}^{+}K\hat{y}\rangle) - 1/2(\langle y^{+}K\hat{y}\rangle + \langle \hat{y}^{+}Ky\rangle)$$

where the averages are over large ensembles of DFT frames. Now under the normalization constraint that

$$y^{+}Ky = 1 \quad \text{and} \quad \hat{y}^{+}K\hat{y} = 1$$

for each raw DFT frame y and estimated frame $\hat{y}$, the error can be put in the form

$$E_{\hat{y}} = 1 - 1/2(\langle y^{+}K\hat{y}\rangle + \langle \hat{y}^{+}Ky\rangle)$$

$$= 1 - 1/2 \sum_{m,n} (\langle y_m^{*}K_{mn}\hat{y}_n\rangle + \langle \hat{y}_m^{*}K_{mn}y_n\rangle)$$

$$= 1 - \sum_{m,n} 1/2(\langle \hat{y}_n y_m^{*}\rangle + \langle y_n \hat{y}_m^{*}\rangle) K_{mn}$$

$$= 1 - \sum_{m,n} \tilde{G}_{nm}K_{mn}$$

or $$E_{\hat{y}} = 1 - \mathrm{tr}\tilde{G}K$$

where $\tilde{G}$ is the Hermitian Cross Spectral Moment Matrix between y and the estimate $\hat{y}$, i.e.,

$$\tilde{G}_{nm} = 1/2(\langle \hat{y}_n y_m^{*}\rangle + \langle y_n \hat{y}_m^{*}\rangle)$$

or

$$G = 1/2(\langle \hat{y}y^{+}\rangle + \langle y\hat{y}^{+}\rangle)$$

2.2.2 <u>Constraint on G and K Induced by $y^{+}Ky=1$</u>

Note that the previously cited normalization constraints

$$y^{\dagger}Ky = 1 \quad \text{and} \quad \hat{y}^{\dagger}K\hat{y} = 1 \ ,$$

applicable to each frame of both raw and of resynthesized data, induce constraints on G, $\hat{G}$ and K of the form

$$\mathrm{tr}\ GK = 1 \quad \text{and} \quad \mathrm{tr}\ \hat{G}K = 1$$

where G and $\hat{G}$ are the spectral moment matrices

$$G = \langle yy^{+} \rangle \quad \text{and} \quad \hat{G} = \langle \hat{y}\hat{y}^{+} \rangle$$

Note the distinction between the spectral moment matrix $\hat{G}$, associated with the estimator $\hat{y}$ and the cross spectral moment matrix

$$\tilde{G} = 1/2(\langle \hat{y}y^{\dagger} \rangle \ + \ \langle y\hat{y}^{\dagger} \rangle)$$

relating y and $\hat{y}$.

Also in the forming the above second moments only actual frequency components will be used in the DFT vectors and not the augmented components created by the zero fill process used in forming the DFT's.

2.2.3 <u>Error Characterization in Terms of the Error Metric Γ and Cross Moment Matrix $\tilde{H}^2$ in the Canonical Coordiante Domain β</u>

Exactly as in Section 2.2.1 and 2.2.2 we have, for the error term $E_{\hat{\beta}}$ relating to the pseudo canonical coordinate vector β, that

$$E_{\hat{\beta}} = 1/2\langle \Delta\beta^{+}\Gamma\Delta\beta \rangle \ = \ 1/2\langle (\beta-\hat{\beta})^{+}\Gamma(\beta-\hat{\beta}) \rangle$$

$$= 1/2(\langle \beta^{+}\Gamma\beta \rangle + \langle \hat{\beta}^{+}\Gamma\hat{\beta} \rangle) - 1/2(\langle \beta^{+}\Gamma\hat{\beta} \rangle + \langle \hat{\beta}^{+}\Gamma\beta \rangle)$$

which, under the normalization constraints

$$= 1 \quad \text{and} \quad = 1$$

reduces to

$$E_{\hat{\beta}} = 1 - \sum_{m,n} \tilde{H}^2_{n,m}\ \Gamma_{m,n}$$

or

$$E_{\hat{\beta}} = 1 - \mathrm{tr}\ \tilde{H}^2\Gamma$$

where $\tilde{H}^2$ is the Hermitian Cross Moment Matrix between β and the estimate $\hat{\beta}$, i.e., $\tilde{H}^2_{nm} = 1/2(<\hat{\beta}_n \beta^*_m> + <\beta_n \hat{\beta}^*_m>$

or $$\tilde{H}^2 = 1/2(<\hat{\beta}\beta^\dagger> + <\beta\hat{\beta}^\dagger>)$$

Also, the normalization constraints cited above induce constraints on H^2, $\tilde{H}^2$ and Γ of the form

$$\mathrm{tr}\, H^2\Gamma = 1 \quad \text{and} \quad \mathrm{tr}\, \hat{H}^2\Gamma = 1$$

where H^2 and $\hat{H}^2$ are the regular second moment matrices.

$$H^2 = <\beta\beta^+> \quad \text{and} \quad \hat{H}^2 = <\hat{\beta}\hat{\beta}^+>$$

As before, $\hat{H}^2$ is not to be confused with the cross moment matrix $\tilde{H}^2$.

2.2.4 Error Characterization on Assumption that K = (1/M)I

Let K be a Euclidean metric in the frequency domain, specified as $K = (1/M)I$ in terms of an M-dimensional window and normalized so that $\mathrm{tr}\, K = 1$.

Then the inner product constraint on y is of the form

$$y^+Ky = \frac{1}{M}\, y^+y = 1$$

and the associated trace constraint, $\mathrm{tr}\, GK = 1$, where $G = <yy^+>$, reduces to $\mathrm{tr}\, G = \mathrm{tr}<yy^+> = M$.

Now letting $\beta = V^\dagger y$ and choosing V as a unitary matrix that diagonalizes K to Γ via the eigenvector decomposition $V^+KV = \Gamma$ yields

$$\Gamma = (1/M)V^+V = (1/M)I \quad ,$$

with

$$\mathrm{tr}\, \Gamma = 1 \quad ,$$

and with no other condition (except unitarity) on V at the point. Hence if we <u>choose V so that it diagonalizes $G = <yy^+>$</u>, then

$$V^+GV = V^+<yy^+>V = <\beta\beta^+> = H^2$$

hence

$$H^2 = \langle \beta\beta^+ \rangle$$

is diagonal. Since $\beta^+\Gamma\beta = (1/M)\beta^+\beta = 1$, it follows that $(1/\sqrt{M})\beta$ is, in this situation, the canonical coordinate z, and $\Lambda = H^+\Gamma H^2$. Also

$$\text{tr } H^2 = \text{tr}\langle \beta\beta^+ \rangle = \text{tr } G = M$$

The error $E_{\hat{\beta}}$ can be expressed as

$$E_{\hat{\beta}} = 1 - \text{tr } \tilde{H}^2\Gamma$$

where $$\tilde{H}^2 = 1/2(\langle \hat{\beta}\beta^+ \rangle + \langle \beta\hat{\beta}^+ \rangle) .$$

However, since $\hat{\beta}$ is a truncated version of β, we have

$$\hat{\beta} = (\hat{\beta}_0, \ldots, \hat{\beta}_{p-1}, 0_p, \ldots, 0_{M-1})^t ,$$

where the components of $\hat{\beta}$ have been renormalized so that $\hat{\beta}^+\Gamma\hat{\beta} = \frac{1}{M}\hat{\beta}^+\hat{\beta} = 1.$

Hence

$$\hat{\beta}_n = (1/c)\beta_n \quad \text{for} \quad n=0,\ldots,p-1 ,$$

where c is a function of both the frame t specifying the vector $\beta(t)$ and also of p. Hence

$$c^2(t) = \sum_{k=0}^{p-1} \gamma_k |\beta_k(t)|^2 = (1/M) \sum_{k=0}^{p-1} |\beta_k(t)|^2 \leq 1 .$$

Then

$$\tilde{H}^2_{nm} = \left\langle \frac{\beta_n(t)\beta_m^*(t)}{c(t)} \right\rangle \cdot \begin{cases} 1 \text{ if both } n \,\&\, m < p \\ 1/2 \text{ if either } n \text{ or } m \geq p \\ 0 \text{ if both } n \,\&\, m \geq p \end{cases}$$

and

$$\text{tr } \tilde{H}^2\Gamma = \sum_{k=0}^{p-1} \left\langle \frac{|\beta_k|^2}{c} \right\rangle \gamma_k = \left\langle \frac{\sum_{k=0}^{p-1} \gamma_k |\beta_k|^2}{c} \right\rangle = \left\langle \frac{c^2}{c} \right\rangle = \langle c \rangle$$

Hence the error can be written

$$E_{\hat{\beta}} = 1 - \text{tr}\ \tilde{H}^2\Gamma = 1 - \langle c(t)\rangle$$

or

$$E_{\hat{\beta}} = 1 - \langle(\frac{1}{M}\sum_{k=0}^{p-1} |\beta_k(t)|^2)^{\frac{1}{2}}\rangle$$

2.2.5 Error Characterization on Assumption that K = G

Another situation of considerable importance occurs when the error metric K, in the frequency domain, is taken to be equal to the long term average spectral moment matrix G, i.e.,

$$K = G = \langle yy^+\rangle\ .$$

Here the inner product constraint on y is of the form

$$y^+Ky = y^+Gy = 1\ ,$$

and the associated trace constraint, tr GK = 1, reduces to

$$\text{tr}\ G^2 = \text{tr}\ \langle yy^+\rangle^2 = 1\ .$$

Now letting $\beta = V^+y$ and choosing V so that it diagonalizes $K = G = \langle yy^+\rangle$ to Γ via the eigenvector decomposition $V^+KV = \Gamma$ yields

$$V^+GV = V^+\langle yy^+\rangle V = \langle\beta\beta^+\rangle = H^2 = \Gamma$$

hence

$$\Gamma = \langle\beta\beta^+\rangle = H^2 \quad \text{with} \quad \gamma_k = \langle|\beta_k|^2\rangle = H_n^2$$

is diagonal, and in particular, since

$$\text{tr}\ \Gamma^2 = \text{tr}\ V^+G^2V = \text{tr}\ G^2 = 1$$

we have

$$\text{tr} \qquad = 1$$

hence

$$\text{tr}\ \Gamma^2 = \text{tr}\ G = \text{tr}\ \langle\beta\beta^+\rangle = 1\ .$$

Note that $\hat{\beta}$ is a true canonical coordinate since $\hat{\beta}^{+}\Gamma\hat{\beta} = 1$ and $<\beta\beta^{+}>$ is diagonal.

The error E can be expressed as

$$E_{\hat{\beta}} = 1 - \text{tr}\, \tilde{H}^2 \Gamma$$

where, as in Section 2.2.4,

$$\tilde{H}^2 = 1/2(<\hat{\beta}\beta^{+}> + <\beta\hat{\beta}^{+}>)$$

However, since $\hat{\beta}$ is a truncated verion of β, we have

$$\hat{\beta} = (\hat{\beta}_0, \ldots, \hat{\beta}_{p-1}, 0_p, \ldots, 0_{M-1})^t ,$$

but, in this situation, $\hat{\beta}$ satisfies the normalization

$$\hat{\beta}^{+}\Gamma\hat{\beta} = \hat{\beta}^{+}H^2\hat{\beta} = 1 \quad .$$

Hence

$$\hat{\beta}_n = (1/c)\beta_n \quad \text{for} \quad n=0,\ldots, p-1$$

where c is a function of both the frame t specifying the vector $\beta(t)$ and also of p. Thus c is not the same as in Section 2.2.4 since now $\gamma_k = <|\beta_k|^2>$,
hence

$$c^2(t) = \sum_{k=0}^{p-1} \gamma_k |\beta_k(t)|^2 = \sum_{k=0}^{p-1} <|\beta_k|^2> |\beta_k(t)|^2 \quad .$$

Then

$$\tilde{H}^2_{nm} = < \frac{\beta_n(t)\beta^*_m(t)}{c(t)} > \cdot \begin{cases} 1 \text{ if both } n \,\&\, m < p \\ 1/2 \text{ if either } n \text{ or } m < p \\ 0 \text{ if both } n \,\&\, m \geq p \end{cases}$$

and

$$\text{tr}\, H^2\Gamma = \sum_{k=0}^{p-1} < \frac{|\beta_k|^2}{c} > \gamma_k = < \frac{\sum_{k=0}^{p-1} \gamma_k |\beta_k|^2}{c} > = <\frac{c^2}{c}> = <c> \quad .$$

Hence

$$E_{\hat{\beta}} = 1 - \mathrm{tr}\, \tilde{H}^2 \Gamma = 1 - \langle c(t) \rangle$$

or

$$E_{\hat{\beta}} = 1 - \langle (\sum_{k=0}^{p-1} \langle |\beta_k|^2 \rangle \, |\beta_k(t)|^2)^{\frac{1}{2}} \rangle$$

2.2.6 Error Relationship Between Assumption that K = I and K = G

In Section 2.2.4, under the assumption that the error metric K was of the simple Euclidean form

$$K = (1/M)\, I \ ,$$

we saw that

$$\Gamma = (1/M)\, I \quad \text{or} \quad \gamma_k = 1/M \text{ for all } k=0, \ldots, M-1$$

hence

$$\mathrm{tr}\, \Gamma = 1$$

and that the error $E_{\hat{\beta}}$ associated with truncated

$$\hat{\beta} = (\hat{\beta}_0, \ \ldots, \ \hat{\beta}_{p-1}, \ 0_p, \ \ldots, \ 0_{M-1})^t$$

could be expressed as

$$E_{\hat{\beta}} = 1 - \mathrm{tr}\, \tilde{H}^2 \Gamma = 1 - \langle c(t) \rangle$$

where

$$c^2(t) = \sum_{k=0}^{p-1} \gamma_k |\beta_k(t)|^2 = \sum_{k=0}^{p-1} (1/M) |\beta_k(t)|^2 \quad 1$$

hence

$$E_{\hat{\beta}} = 1 - \langle (\sum_{k=0}^{p-1} (1/M) |\beta_k(t)|^2)^{\frac{1}{2}} \rangle$$

And in Section 2.2.4, under the assumption that the error metric K was given by the long term average spectral moment matrix $G = \langle yy^{\dagger}\rangle$, i.e.

$$K = G = \langle yy^{\dagger}\rangle$$

we saw that

$$\Gamma' = H^2 = \langle \beta\beta^{\dagger}\rangle \quad \text{or} \quad \gamma_k' = H_k^2 = \langle|\beta_k|^2\rangle \quad \text{for all}$$

$k = 0, \ldots, M$,

hence

$$\mathrm{tr}\,\Gamma'^2 = 1 \quad \text{and} \quad \mathrm{tr}\,\Gamma' \geqq 1 \quad ,$$

and that the error $E'_{\hat{\beta}}$ associated with truncated $\hat{\beta}$ as above could be expressed as

$$E'_{\hat{\beta}} = 1 - \mathrm{tr}\,\tilde{H}^2\,\Gamma' = 1 - \langle c'(t)\rangle$$

where

$$c'^2(t) = \sum_{k=0}^{p-1} \gamma^1|\beta_k(t)|^2 = \sum_{k=0}^{p-1} \langle|\beta_k|^2\rangle\,|\beta_k(t)|^2$$

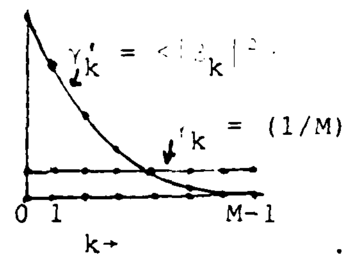
hence

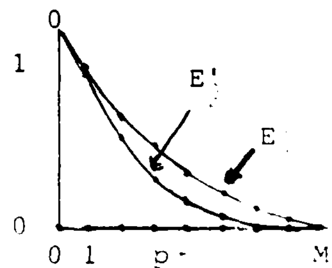
$$E'_{\hat{\beta}} = 1 - \langle(\sum_{k=0}^{p-1} \langle|\beta_k|^2\rangle\,|\beta_k(t)|^2)^{\frac{1}{2}}\rangle \quad .$$

Since the only difference between $E_{\hat{\beta}}$ and $E'_{\hat{\beta}}$ is that $\gamma_k = 1/M$ is flat for all k and $\gamma_k' = \langle|\beta_k|^2\rangle$ is monotonically decreasing for increasing k, with $\mathrm{tr}\,\Gamma = 1$ and $\mathrm{tr}\,\Gamma' \geqq 1$, we see that

$$E'_{\hat{\beta}} \leqq E_{\hat{\beta}}$$

for each level of truncation p.





Thus for any given level of truncation p the error derived by using an error metric based on the spectral moment matrix $G = \langle yy^{\dagger}\rangle$ will generally be significantly less than the error based on the use of a simple Euclidean error metric.

2.2.7 Error Metric Considerations in the Psychoacoustic Domain

In Section 2.2.6 we saw that an error metric K, based upon the use of the long term spectral moment matrix $G = \langle yy^{+} \rangle$ in the frequency domain, rather than a simple Euclidean metric $K = (1/M)I$, will generally lead to a significantly smaller error, for any given level of truncation in the pseudo-canonical coordinate representation β, of the signal to be transmitted.

This error reduction is achieved by making use of the second order statistics of the signal in the frequency domain. And a precisely similar result can be achieved by utilizing the long term cross correlation matrix $R = \langle xx^{+} \rangle$, in the time domain, for the error metric J.

Thus far the above approach has only taken psychoacoustic information into account to the extent that it is reflected in the long term first and second moment statistics of the signal generation process. This process is probably not insignificant to the extent that the vocal tract speech production process can be configured to produce sounds having a reasonably higher statistical probability of being discriminated by the auditory system against background of various kinds of noise and other types of acoustical interference.

The approach outlined does, however, offer certain possibilities for applying some systematic variational techniques to obtain from the metric K=G a new metric $\tilde{K}=\tilde{G}=1/2(\langle y\hat{y}^{+}\rangle + \langle \hat{y}y^{+}\rangle)$ that takes into account both the spectral moment statistics and also the cross spectral moment statistics between the raw spectral signal y in the frequency domain and series of estimates $\hat{y}=V\hat{\beta}$ that are determined by a signal resynthesis and listening process.

2.2.8 Psychoacoustic Comparison of Error Metric Assumptions K=I and K=G

Since the use of relatively simple statistical error minimization criteria, such as those embodied in LPC-based short term Euclidean error minimization techniques, have proved quite psychoacoustically successful in narrow band voice signal representation systems, a reasonable initial assumption to test is the psychoacoustic improvement that might be achieved in utilizing the somewhat more complex statistical error minimization criterion based on the spectral moment error metric $K=G=\langle yy^{+} \rangle$, as compared with the simpler Euclidean error metric $K=(1/M)I$.

We have seen in Section 2.2.6 that the actual statistical error $E_{\hat{\beta}}$, based on the metric K=G, should be smaller than the error $E_{\hat{\beta}}$, based on the metric K=(1/M)I, for each level of truncation p of the pseudo=canonical coordinate representation. And it is reasonable to expect, as noted in Section 7, that the human auditory system can also make use of additional statistical information, such as that embodied in the metric K=G, in concluding that the psychoacoustic performance of this type of system is superior to that of the Euclidean metric based system, for each level of truncation p.

A relatively simple set of A-B comparison tests, run both within and between voice samples from the two different systems, at four to six different levels of truncation, should serve to provide a rough quantation comparison of the relative performance of the two systems, and hence of the value of the additional statistical information embodied in the error metric K=G.

2.3 CANONICAL COORDINATE VOCODER ALGORITHM IMPLEMENTATION AND EXAMPLE

The CC decomposition algorithm and routines to generate spectral moment error metrics have been implemented at the RADC/EEV Speech Processing Facility. Coding is accomplished using a combination of Fortran and APAL for the PDP-11/44 computer and the FPS AP-120B array processor. Routines run in non real-time and utilize the Speech Data Base (see Chapter 6) for data I/O. The program CCVOC will accept any defined error metric, transform input speech to the CC domain, compress the data by an amount called for and generated an output file of synthesized speech. No coding or quantization is performed by this routine at this time. A separate program EIGEN calculates the eigenvalues and eigenvectors for the error metric being considered and generates a file with the information required by CCVOC for a specific error metric. Other routines have been implemented to generate statistically and empirically defined error metrics.

<u>CC Error Metric Eigensolution</u> - To illustrate the CC algorithm, a non-Euclidean error metric has been defined statistically from speech using the long term cross spectral moment method discussed in Section 2.2. This is an example of an error metric and is not representative of the "best" error metric for any given system. The error metric is presented graphically in Figure 2.2 as a 3-dimensional representation of the 64 by 64 averaged spectral moment matrix. The hermitian characteristic of the matrix can be seen in this figure. The first step in the CC decomposition process is to calculate the eigenvalues for this matrix and the associated eigenvectors. This is accomplished by the routine EIGEN and the results are shown in Figure 2.3. The solution of the eigensystem requires that the input matrix be Hermitian. The eigenvectors combine to define the transformation matrix V. The diagonalization of the error metric in the CC domain (Γ') is evident as it is now fully defined by its diagonal elements, the eigenvalues.

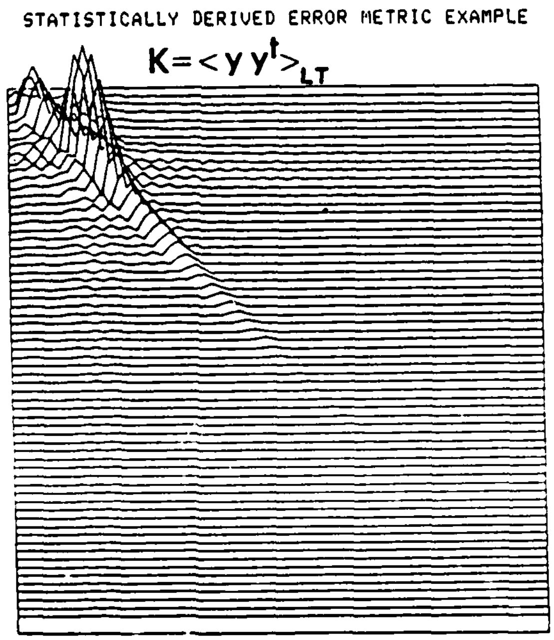


Figure 2.2 Spectral Moment Error Metric

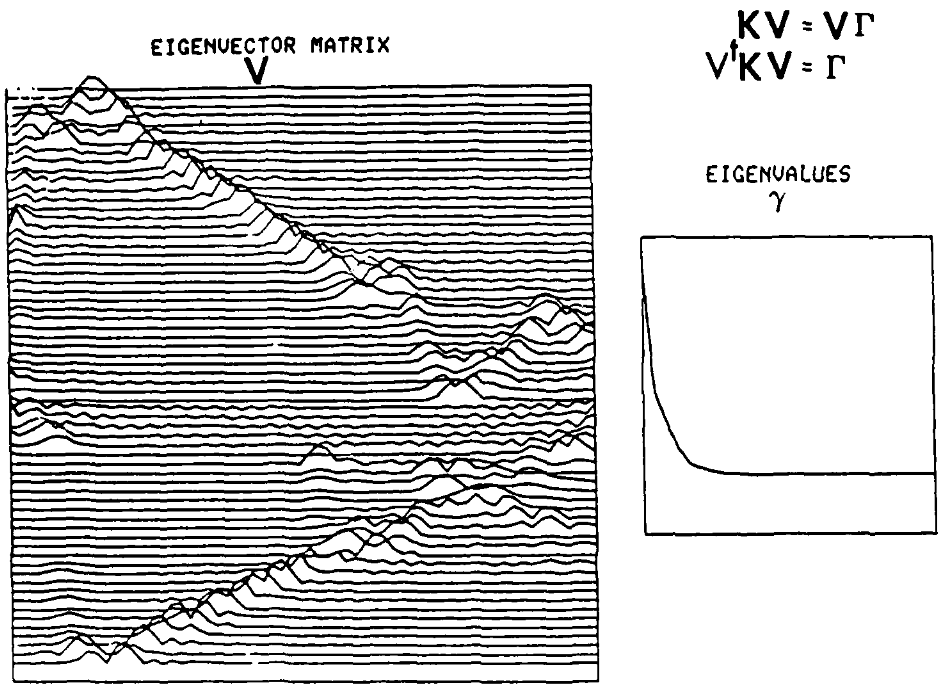


Figure 2.3 Error Metric Eigensolution

Speech Analysis - Sampled speech is now input to the algorithm in a frame by frame manner. The input speech can be preprocessed to provide for preemphasis, windowing and signal normalization. The speech frame is zero filled to twice its length and the Discrete Fourier Transform (DFT) operator Z is used to go to the frequency domain as shown in Figure 2.4. The complex conjugate transpose (†) of the CC transformation matrix V is now used to take the frequency domain signal y into the pseudo canonical coordinate domain as shown in Figure 2.5.

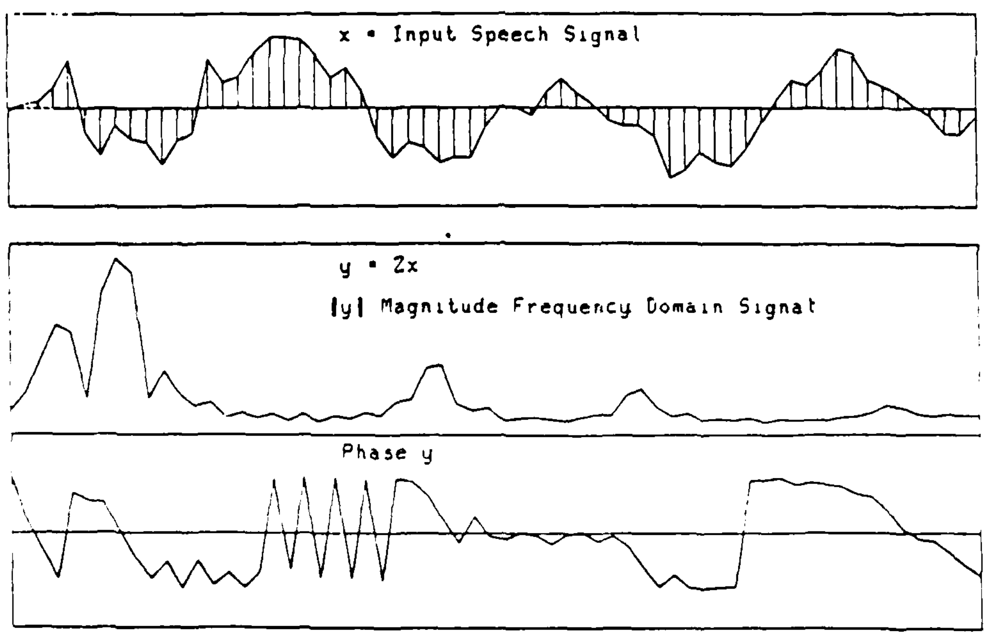


Figure 2.4 Initial CC Algorithm Processing

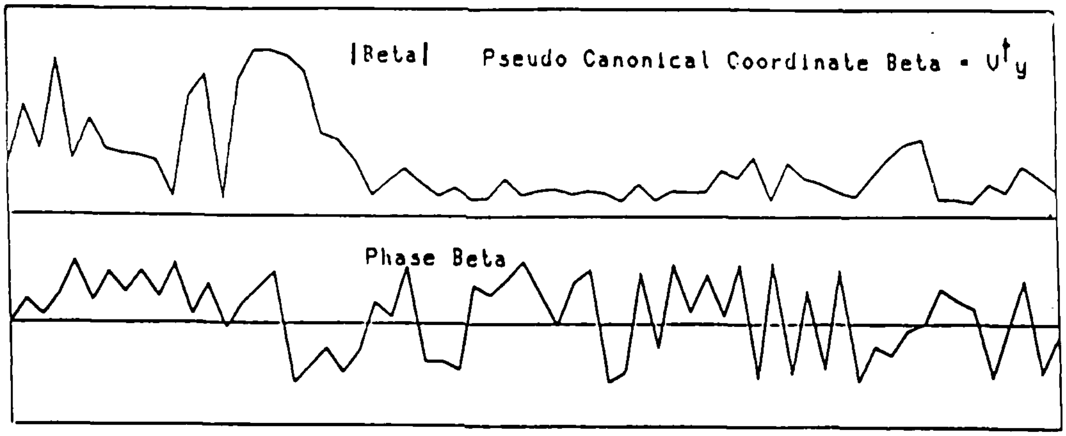


Figure 2.5 Pseudo CC Signal Transformation

At this time an ordering procedure must be defined that will relate the pseudo CC domain to the CC domain. This is accomplished by calculation the diagonal of the CC transformation H, which can be shown to be equal to the "power spectrum" of the signal in the pseudo CC domain. This vector is then weighted by the eigenvalues and ordered on magnitude. This process provides the diagonal signal correlation requirement of a true CC transform. Figure 2.6 shows the ordering process, while Figure 2.7 gives the permutation vector P and demonstrates how it is used to reorder the pseudo CC signal vector. The permutation vector is just an array of indirect address offsets. The important fact to remember about the reordered signal of Figure 2.7 is that its components are now ordered by importance relative to the error minimization defined by the error metric used. Truncation of these parameters from the left, one by one, is guaranteed to result in minimum synthesis error.

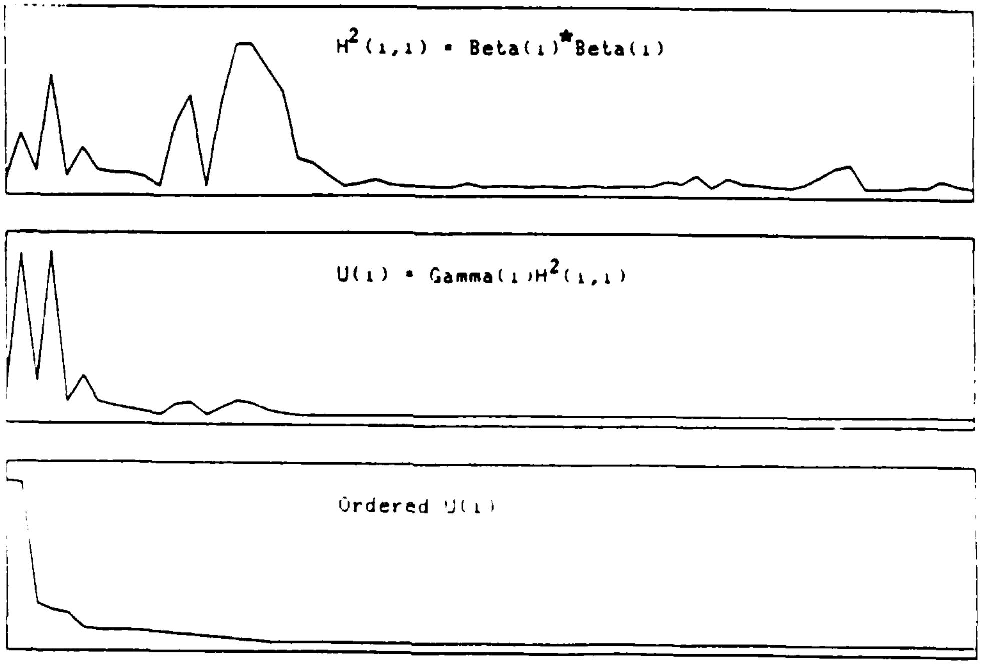


Figure 2.6 CC Ordering Procedure

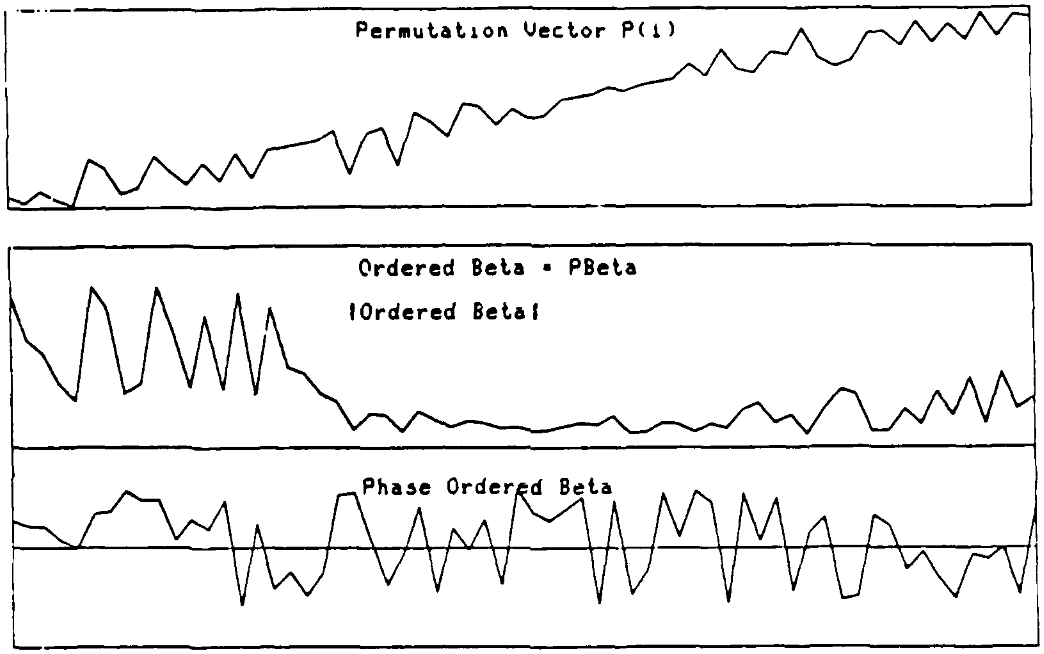


Figure 2.7 Reordered CC Signal

Signal Compression And Reconstruction - Signal reconstruction starts with the transmission of parameters from the analyzer to the synthesizer. The number of parameters, their quantization and coding define the final bit rate for a given system. For each parameter transmitted, its complex parts and a permutation value must be included. Since the signal parameters at the analyzer have been ordered relative to their importance to synthesis error as defined by the error metric, it can be seen that the reduction of the number of parameters to be transmitted has been simplified.

Figure 2.8 demonstrates the algorithm steps for signal reconstruction without compression. It should be noted that Figures 2.8 - 2.10 only the magnitude values for the pseudo CC and frequency domain signal; however, the reconstruction process does handle both the magnitude and phase of the signal. All complex parameters and the full permutation vector have been transmitted to the synthesizer. The pseudo CC signal is reordered using the permutation vector. The transformation matrix V is then used to take the signal estimate into the frequency domain y. The inverse DFT operator generates a time domain signal estimate that is twice the length of the original signal. After the last half of this signal is striped off, the signal renormalized and deemphasized if necessary, a reconstructed estimate of the original framed data results. A comparison of Figures 2.4 and 2.8 shows exact reconstruction for the case of no compression as expected.

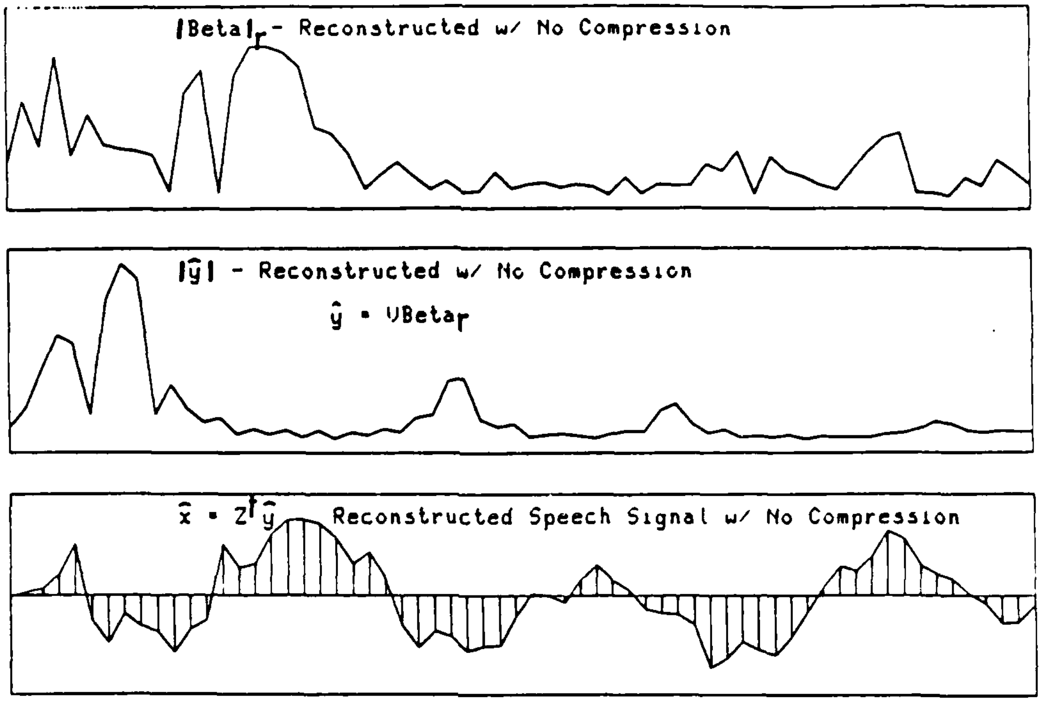


Figure 2.8 CC Reconstruction Procedure

Figures 2.9 and 2.10 give reconstruction of the same signal with 60% and 90% of the ordered pseudo CC parameters eliminated. This elimination of some percentage of the parameters is defined as a percent compression for later use. It should not be confused with an actual data transmission rate. At the synthesizer, the received parameters are reordered using the permutation values and missing parameters are set equal to zero. The distribution of these a priori zeroed parameters by the reordering process can be seen in these figures. A definite smoothing of the reconstructed time domain signal is evident at the 60% level, while at 90% high frequency components have been introduced. A comparison of the frequency domain magnitudes for the three reconstruction examples demonstrates how "extra" information is forced into the spectra by the V transformation matrix. It is important to emphasize that the error metric used for this example does not necessarily provide the best signal reconstruction. This error metric was chosen to demonstrate the CC process for a non-Euclidean error criteria. The "best" error metric for a given vocoder system has yet to be defined.

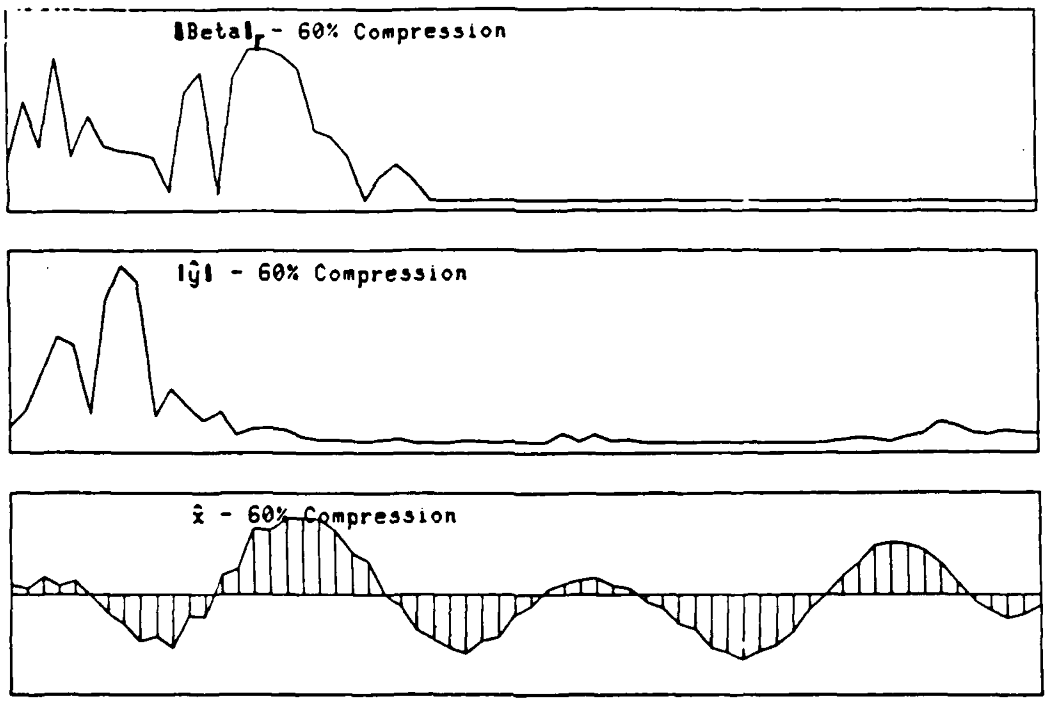


Figure 2.9 Sixty Percent Compression Example

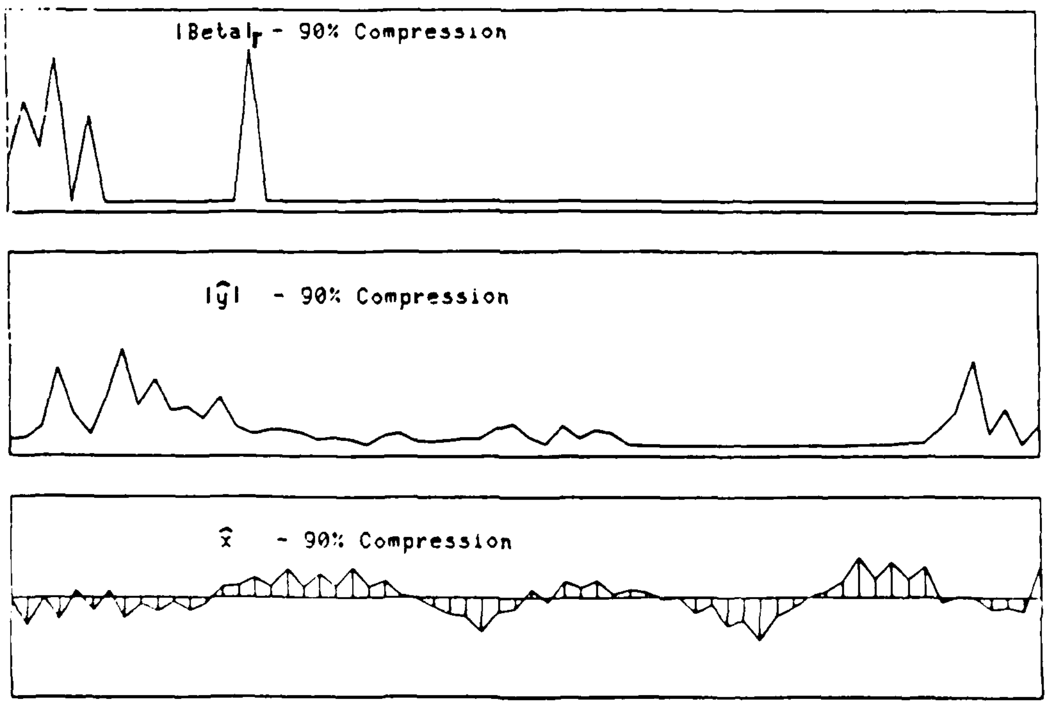


Figure 2.10 Ninety Percent Compression Example

2.4 ERROR METRIC AND CC VOCODER EXAMPLES

Several error metrics have been generated to date in order to experiment with the versatility of the vocoder program CCVOC and with the effects of various approaches to the definition of an error metric. This work has ranged from the simplistic example of the identity matrix I as an error metric to metrics based on statistical characteristics of the speech signal and empirically defined metrics that model channel vocoders.

2.4.1 Vocoder Signal Processing

Principal Component Vocoder Example - If the identity matrix I is used as the error metric, the Euclidean or least squares error minimization criteria results. This reduces the CC method to the well known principal component analysis method. The ordering procedure inherent in the CC method is now based on the size of the power spectral components, and compression consisting of eliminating those spectral components with the least power. Reconstruction examples at five compression values (20%, 60%, 80%, 90%, 95%) for five frames of speech are given in Figure 2.11 along with the unprocessed speech signal. As the percent compression increases the frame effects become evident in the reconstructed signal. The smoothing of the signal as high frequency components are dropped out can be clearly seen; however, it is not until compression steps from 90% to 95% compression that most structure in the signal is lost. To place this in perspective the reader must remember that at 90% only six spectral components are being used and at 95% there are only 3 components available.

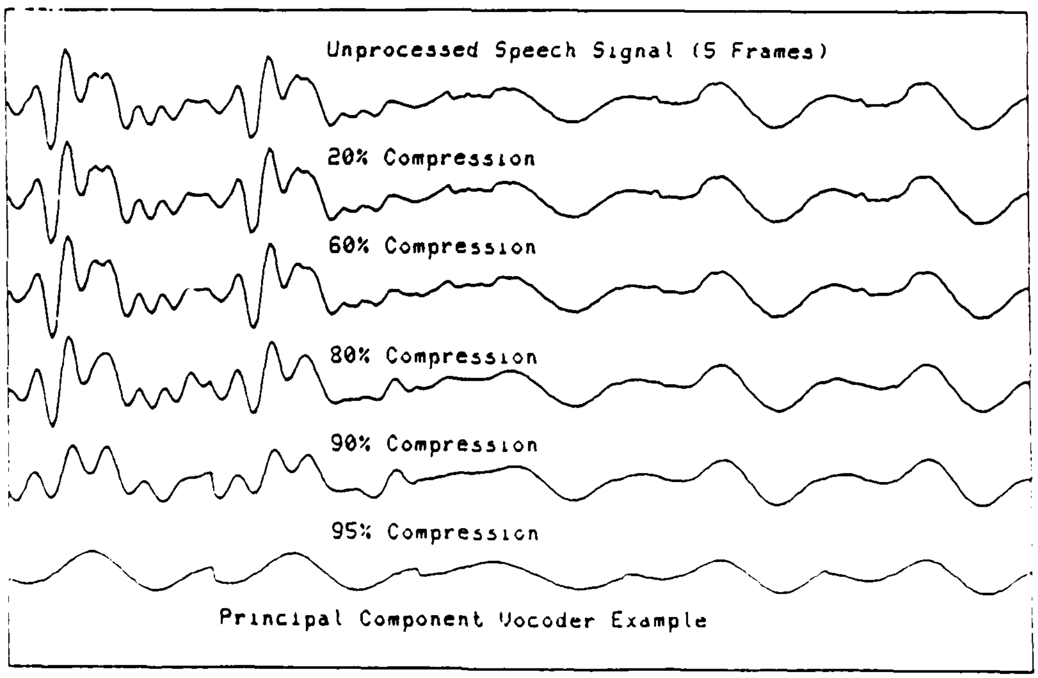


Figure 2.11 Principal Component Vocoder Example

Spectral Moment Vocoder Example - A non-Euclidean error metric that incorporates the statistical redundancies of speech can be based on either ensemble averages of the spectral moment matrix in the frequency domain, or equivalently, the cross correlation matrix in the time domain. This candidate has certain useful features. It makes optimum use of the spectral second moment statistics in providing a minimum quadratic error. Since it is reasonable to assume that the human auditory system has some capability for making use of this second moment information in the speech signal discrimination process, it follows that this metric should form a suitable basis for the development of a metric that defines some of the more elusive error criteria of speech. Such an error metric has been introduced as Figure 2.2. The development and use of this metric has indicated the importance of the averaging process and speech material used to generate the metric.

This error metric has been used with CCVOC with limited results. Reconstruction examples at five compression values (20%, 60%, 80%, 90%, 95%) for five frames of speech are given in Figure 2.12 along with the unprocessed speech signal. As in the algorithm example of Section 2.3, high frequency components are forced into the spectra of the synthesized signal. This is especially evident above 60% compression. Real time analysis of the output speech on the RADC/EEV SD350 spectral analyzer shows a definite loss of formant motion in the speech. This seems to result in a smearing of speaker dependent characteristics. The input speech that defined the spectral moment error metric consisted of DAM sentences from three male speakers. The reconstruction of a DRT word list from one of these speakers shows a freezing of the formant locations at compression ratios above 60%. This can be traced to the average of the auto-spectra for the speech used to generate the error metric. The "average" formant locations defined in the error metric now control their positions at compression ratios above 60%. This is especially true above the second formant.

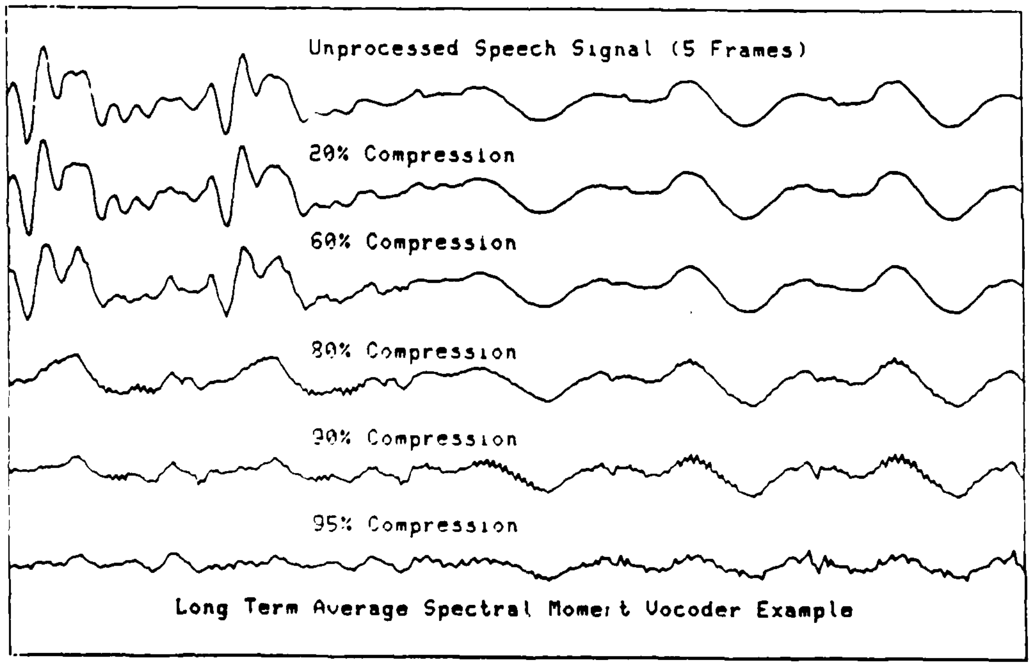


Figure 2.12 Spectral Moment Vocoder Example

Sixteen Channel Vocoder Example - The traditional analog channel vocoder was designed to take advantage of several known psychoacoustic characteristics of the human auditory system. These include the logarithmic decrease of sensitivity of the hearing process with increasing frequency and the broadening of bandwidth at higher frequencies. As a first attempt to incorporate some psychoacoustic effects into our error minimization process, it was decided to develop an empirical error metric that modeled these affects. It became apparent after several failures in creating a frequency domain error metric, that this can be best accomplished by defining the eigensystem solution itself. Figure 2.13 gives the empirical definition of a "sixteen channel vocoder" with logarithmicly spaced center frequencies and increasing bandwidth. It consists of 16 rectangular non-overlapping filters that span the full spectral range of the analysis. This definition leaves 75% of the channels undefined and sets all but the first 16 eigenvalues equal to zero. The eigenvalues and "filter" areas have been normalized for the 16 channels that are defined. Since all channels are not defined, this error metric differs from the previous ones introduced in that it can not provide full signal reconstruction.

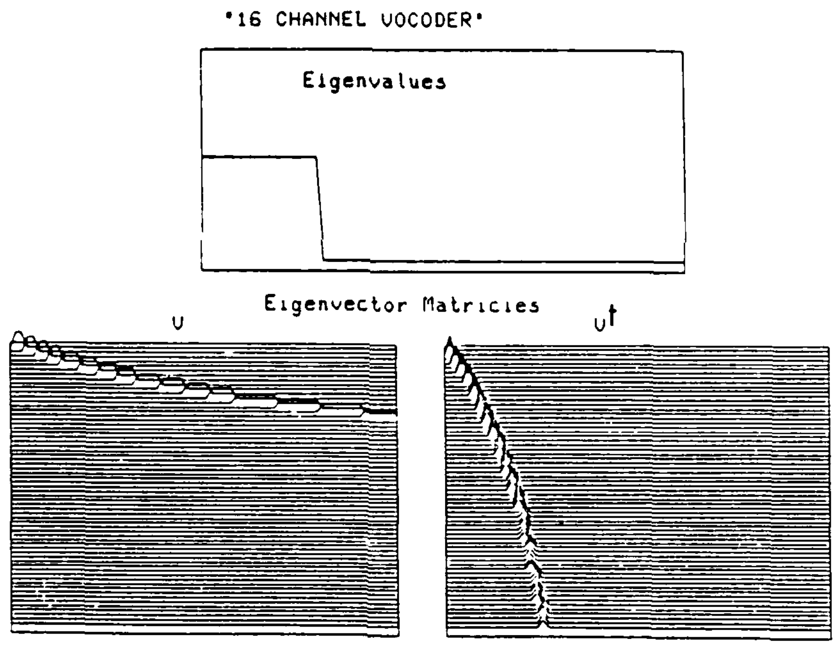


Figure 2.13 "16 Channel Vocoder" Eigensolution

When the preceding error metric is used by the program CCVOC, some very interesting results are achieved. Reconstruction examples at five compression values (0%, 75%, 80%, 90%, 95%) for five frames of speech are given in Figure 2.14 along with the unprocessed speech signal. Since the error metric is not fully defined it is evident that the unprocessed signal and the 0% compression signal are markedly different. Since 16 of 64 channels are defined, the 0% signal estimate matches the 75% compression signal. Frame boundaries can be seen at 90% and above; however, the structure of the signal estimate at 95% compression is quite representative of the input signal and is considerably better that and of the previous examples.

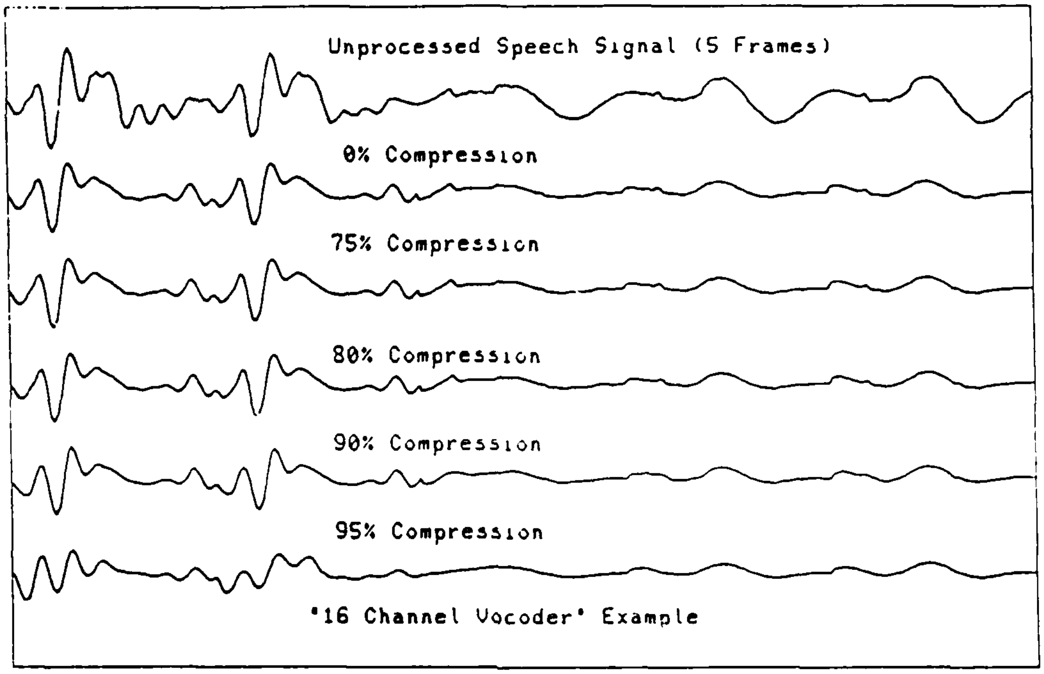


Figure 2.14 "16 Channel CC Vocoder Example"

2.4.2 Vocoder Comparisons And Intelligibility Testing

All of the preceding vocoder examples run using the program CCVOC on the PDP-11/44 & AP120-B combined processors at or under 20 times real time. This and the Speech Data Base capability at RADC/EEV has allowed us to process Diagnostic Rhyme Test (DRT) tapes for evaluation of the intelligibility of the different systems and demonstration sentence sets. Testing of three male speakers in a quiet environment has been completed for the examples presented. Figure 2.15 shows the synthesized speech signal for the sentence "Tom's birthday is in June" as processed by the three vocoders and the "unprocessed" 96 kbps PCM 4 kHz lowpass filtered input signal. All of the vocoders are operating at a 90% compression ratio. The total DRT scores averaged over four repeats of the test are also given in this figure.

Intelligibility testing of these systems was performed to give us an indication of the direction to take in future research on optimum error metric definition. These scores are not meant to be used as a rating of the CC method. Detailed DRT results for these systems are presented in Table 2.1 and Figure 2.16.

The scores for the spectral moment vocoder indicate that development of statistically defined error metrics will be limited by the short term time stationarity of the speech signal. The comparatively good scores for the "16 Channel Vocoder" when compared to the "unprocessed" and Principal Component scores are promising. They indicate that an empirical definition of known psychoacoustic affects can result in a non-Euclidean error minimization criteria with high intelligibility. This is especially evident when you note that the reconstructed signal is quite different from the unprocessed or Principal Component signal; however, the intelligibility is still high.

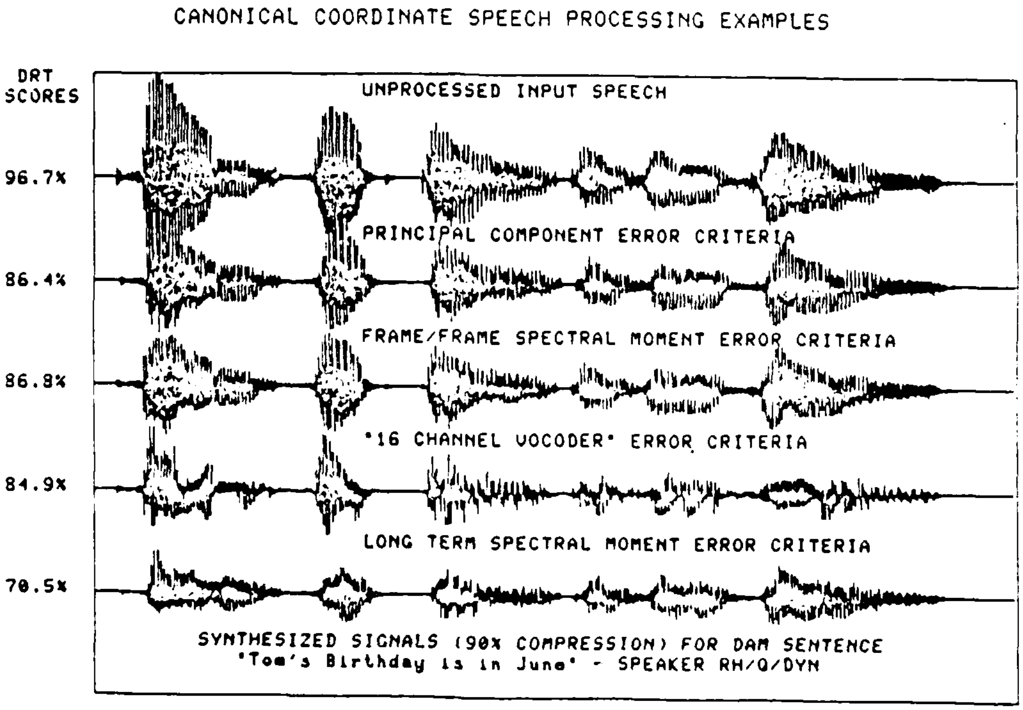


Figure 2.15 Canonical Coordinate Speech Processing Examples

DRT results from a 2.4 kbps LPC-10, a 16 kbps CVSD and a 9.6 kbps APC for the same male speakers in the quiet are included in Figure 2.16. It is difficult to define a comparative data transmission rate for our CC vocoder examples. Although a compression ratio is set, there is no parameter quantization or coding involved. We do feel that it is fair to equate the 90% compression factor with a transmission rate some where between 2.4 and 9.6 kbps. Analysis of the DRT data shows that the psychoacousticly based "channel vocoder" has difficulty with the speech attributes sustention and graveness. These are the attributes with which many speech compression systems have problems. A close look at the data shows that it is the recognition of the absent state of these attributes that is the main problem. This means that a listener may hear cheat as sheet or tong as thong (sustension); thought as fought or did as bid (graveness). Sustention-absent or "interrupted" correlates to an abrupt onset of energy across the the full spectrum with a duration usually less than 130 msec. Graveness-absent correlates to a high location of the second and third formants with a resulting concentration of energy in the upper half of the spectrum. These facts gives us some indication of what may be missing from this error metric. Work is continuing to define an optimized non-Euclidean error criteria.

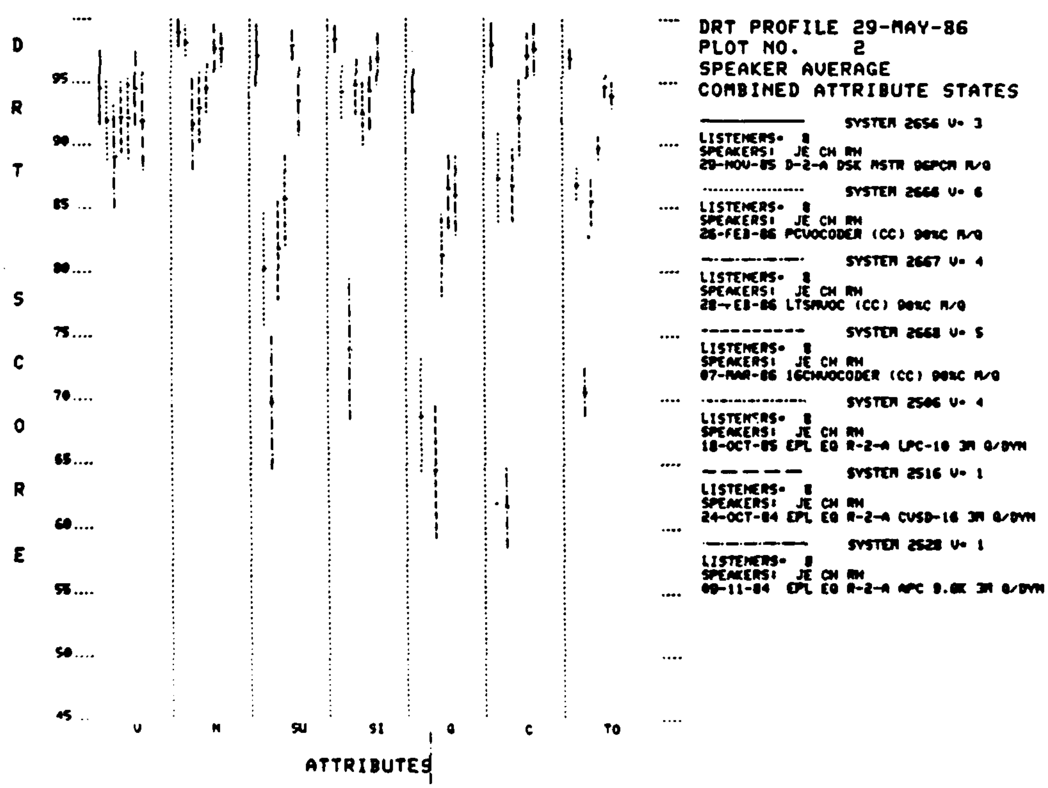


Figure 2.16 CC Vocoder Intelligibility

DRT SUMMARIES 29-MAY-86

SPEAKER AVERAGE

		FEATURES						
		V	N	SU	SI	G	C	TOTAL
SYSTEM 2656 V= 3								
P+A:	MEAN	94.53	98.96	97.14	98.44	94.27	97.92	96.88
	S.E.	1.52	0.61	1.25	0.56	0.92	0.90	0.42
P:	MEAN	96.87	99.48	94.79	97.40	92.71	97.92	96.53
	S.E.	1.13	0.52	2.37	1.06	1.67	0.97	0.45
A:	MEAN	92.19	98.44	99.48	99.48	95.83	97.92	97.22
	S.E.	2.90	1.14	0.52	0.52	1.44	0.97	0.66
SYSTEM 2666 V= 6								
P+A:	MEAN	91.93	98.18	80.21	94.27	68.75	87.24	86.76
	S.E.	1.74	0.70	2.37	1.24	2.32	1.90	0.71
P:	MEAN	97.92	98.44	69.79	92.19	81.77	92.19	88.72
	S.E.	0.97	0.86	4.05	1.26	2.38	1.96	0.98
A:	MEAN	85.94	97.92	90.62	96.35	55.73	82.29	84.81
	S.E.	3.22	1.23	2.29	1.76	5.10	3.09	0.88
SYSTEM 2667 V= 4								
P+A:	MEAN	89.06	91.67	69.79	73.96	37.50	61.72	70.62
	S.E.	2.10	1.91	2.65	2.81	3.36	1.61	0.96
P:	MEAN	91.67	97.40	80.21	64.58	50.52	65.10	74.91
	S.E.	2.08	1.06	3.44	4.97	5.45	4.39	1.16
A:	MEAN	86.46	85.94	59.37	83.33	24.48	58.33	66.32
	S.E.	3.09	4.07	4.89	2.78	7.24	4.17	1.83
SYSTEM 2668 V= 5								
P+A:	MEAN	92.19	92.97	81.77	94.79	64.58	86.72	85.50
	S.E.	1.47	1.47	2.06	1.23	2.68	1.47	1.01
P:	MEAN	94.79	94.79	85.42	94.27	70.31	95.31	89.15
	S.E.	1.49	1.67	3.50	1.68	4.11	1.47	1.18
A:	MEAN	89.58	91.15	78.12	95.31	58.85	78.12	81.86
	S.E.	2.34	2.96	3.03	1.81	4.22	2.74	1.31

SPEAKER AVERAGE

SYSTEM 2656 V= 3	29-NOV-85 D-2-A DSK MSTR 96PCM M/Q NUMBER LISTENERS = 8 SPEAKERS ARE JE CH RH
SYSTEM 2666 V= 6	26-FEB-86 PCVOCODER (CC) 90%C M/Q NUMBER LISTENERS = 8 SPEAKERS ARE JE CH RH
SYSTEM 2667 V= 4	28-FEB-86 LTSMVOC (CC) 90%C M/Q NUMBER LISTENERS = 8 SPEAKERS ARE JE CH RH
SYSTEM 2668 V= 5	07-MAR-86 16CHVOCODER (CC) 90%C M/Q NUMBER LISTENERS = 8 SPEAKERS ARE JE CH RH

Table 2.1 CC Vocoder Intelligibility

CHAPTER 3

ACOUSTIC NOISE CHARACTERIZATION

3.1 INTRODUCTION

A major problem with narrowband digital voice processors is the degradation of their performance by background acoustic noise. Digital voice communication systems utilized by the Air Force are required to operate on a large variety of military platforms. The acoustic noise environment is a function of the specific platform and the operational mode of the platform. Various noise reduction and suppression methods have been tried to solve this problem for specific platforms and processors. Until recently, no research has been directed at characterizing and categorizing the broad range of noise environments of interest to the Air Force, as they affect narrowband voice processors and noise reduction techniques. The research reported in this section surveys the varieties of acoustic noise problems to be found in those environments, using the acoustic noise data library of the RADC/EEV Speech Processing Facility.

3.2 ACOUSTIC NOISE AND SPEECH SIGNALS

In discussing the acoustic noise problem for speech communication systems, it should first be observed that there are a number of determinants of background noise other than the basic noise environment due to the aircraft itself. Among these influences are interfering speech from other speakers; continuous or transient noise from equipment (especially communications equipment) present in the aircraft; unpredictable noise due to firing of weapons; and the effects of oxygen masks, microphones, and noise reduction processing. But before these factors are taken into account, it is necessary to understand the aircraft noise background underlying all these other factors. Our emphasis here is not on trying to characterize acoustic noise completely as a deterministic function of all the operational variables, but rather to examine the range of acoustic noise phenomena facing speech communication systems.

Before the acoustic noise and the acoustic speech signal are processed by narrowband speech systems, they are influenced by other factors. In Section 3.5 we discuss the role of noise cancellation and noise subtraction. We should not forget that the absolute level of the speech itself is controlled by the speaker. A further factor that we have not measured is the noise-suppression effect of oxygen masks. In aircraft such as the F-15 (see Section 3.4), the microphone is inside the oxygen mask, and the mask itself provides much-needed attenuation of the aircraft's acoustic noise.

The overall acoustic noise power is probably the most often quoted attribute of an aircraft's acoustic noise environment. However, for the processing of speech against this noise background, other attributes of the noise may be more significant. Here we distinguish between time-domain and frequency-domain characteristics of the environment.

For speech processing, time-domain characteristics of the background acoustic noise become important. As we discuss in Section 3.5 of this report, some methods of noise reduction are more sensitive than others to variations in the background noise over time. We have chosen to distinguish between two kinds of time-variation in the noise:

1. Long-term variation
2. Short-term variation

Our distinction is between noise that varies slowly (say from minute to minute, or from one flight configuration to another) and noise that varies rapidly, even from one speech analysis frame to the next. These variations in noise over time may be observed in a variety of ways. Long-term variations can be isolated by comparing power spectrum averages, while short-term variations are indicated by "real time" power spectra and the broadening of peaks in averaged power spectra. (See Section 3.3 for a general description of the analysis methods used for this study.) While both types of variation can be significant for speech processing, this study concentrates on long-term variation.

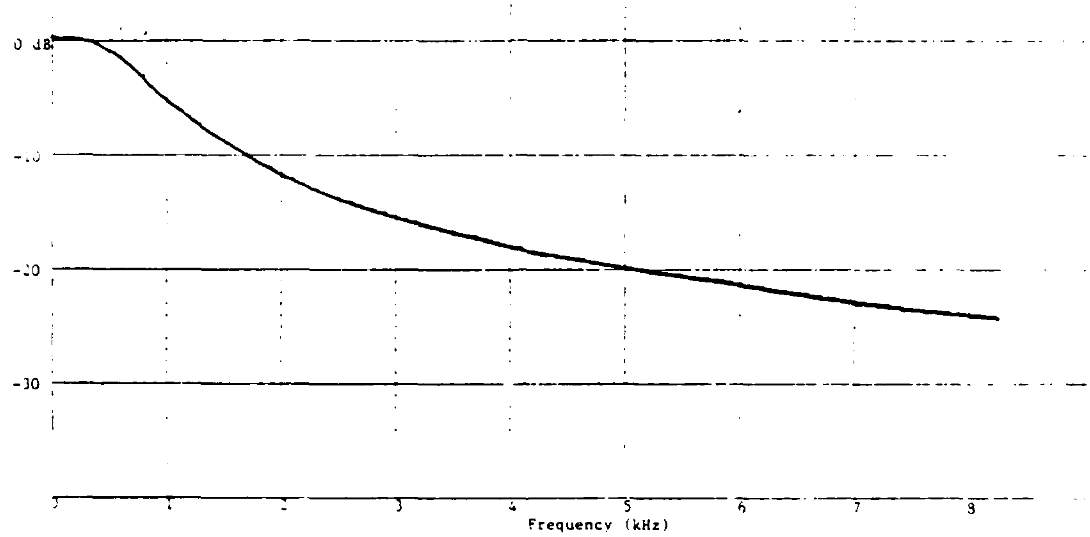
As for the frequency-domain characteristics of the acoustic noise background, we first observe that speech processing is less likely to be corrupted by noise outside the passbands of the analog anti-alias and high-pass filters commonly employed in analyzers. Thus, for narrowband systems, the frequency range of prime importance extends from 100 Hz to about 4000 Hz. Noise outside this range could affect a listener located in the aircraft, but would not directly corrupt an analyzer's voice processing itself.

For the purposes of this research, we have chosen to group the frequency-domain attributes of acoustic noise under three headings:

1. Broad spectral shape
2. "Formant-like" resonant bands
3. Discrete periodic components

By "broad spectral shape" we mean the general shape of the acoustic noise power spectrum, including any overall slope maintained across a large part of the frequency range. To assess the impact of acoustic noise on speech processing, frequency-domain properties of acoustic noise must be compared with the frequency-domain properties of speech. Although speech is inherently highly variable in its spectral shape, the physical shape of the human vocal tract produces a long-term average spectral distribution that is not flat. In fact, one motivation for the preemphasis typically applied as the first stage of digital speech processing is to compensate for the decline in typical speech energy above about 500 Hz. Although this "average shape" is to some extent dependent on the individual speaker, we have followed Oppenheim and Lim (Ref. 3.1) in using the shape shown in Figure 3.1 as a rough average over all speakers. This spectral shape is flat from 100 Hz to 500 Hz and then declines 6 dB per octave above 500 Hz. In discussing acoustic noise spectra later, we shall compare the spectral shape of the noise with this long-term "average" distribution of speech energy.

Fig. 3.1: Average Spectral Distribution of Speech



We use the term "formant-like" to refer to wide-shouldered peaks in the magnitude spectrum extending over a number of frequency bins, and therefore not due to periodic acoustic noise. Such features may be the acoustic output of a resonator passing a narrow band of frequencies, and therefore can masquerade as a speech formant. In other cases, apparent widened peaks may be an artifact of long-term averaging, resulting when time-averaging is applied to spectra containing a time-varying sinusoid sweeping through a range of frequencies. This latter kind of variation would not be seen as widened peaks by sequential or frame-oriented voice processors, which do not perform long-term averaging of the input signal.

Finally there are the relatively well-behaved periodic components of the noise background. If these components are well separated, they may be susceptible to certain noise-reduction processing techniques (as discussed in Section 3.5), but they may pose a special problem to strategies that assume background noise is Gaussian.

3.3 SOURCES OF DATA AND ANALYSIS METHODS

As we have already remarked, the background noise in operational aircraft may be separated into two major components. On one hand there is what we will call "inherent" noise of the aircraft, arising from

1. Turbulent airflow and mechanical vibration associated with the engines, turbines, and propellers;
2. Turbulent airflow around the rest of the aircraft;
3. Vibration of the aircraft's structure excited ultimately by (1) and (2) above.

This "inherent" noise is the noise arising because the aircraft is flying in a certain control configuration through a certain external aerodynamic environment. Contrasted with "inherent" noise is noise arising from operations within the aircraft, such as the acoustic noise caused by weapons, communications equipment, or other speakers. This study does not attempt to predict or classify the effects of such "operational" noise sources.

Narrowband speech processing must cope with a range of inherent noise environments occurring during aircraft operations. Other studies directed toward the development of particular noise control methods have focused specifically on a tightly constrained range of noise environments. The goal of this study, on the other hand, has been to characterize the entire range of inherent aircraft noise environments present in the RADC Speech Processing Facility's acoustic noise data base. This data base consists of acoustic noise recordings made with high-quality microphones and recording equipment aboard several aircraft of different types (Table 3.1). In some cases, these recordings were made at more than one location within an aircraft; in other cases, an attempt was made to record noise in a variety of operational flight configurations. Most of the recordings are well enough documented to allow inference of the absolute sound levels present during recording.

For this study the primary analysis tools were the real-time spectral analyzers (SD360 and SD350) at the EEV Speech Processing Facility. The Spectral Dynamics SD360 real-time spectrum analyzer (Ref. 3.2), with built-in anti-alias filters and A/D converters, is capable of a number of dual-channel operations including FFT-based operations on ensembles of 1024 or 2048 data points. The SD360's repertoire of functions includes autocorrelation, cross spectrum, and probability density analysis, and averaging of multiple input ensembles in the time or frequency domain. The capture of transient signals is also provided for. This instrument is interfaced to the PDP-11/44 system with a software package called "AGP" (Automated Graphics Package) that allows analysis results to be read out, plotted on the Tektronix display, and stored on disk files (Ref. 3.3). AGP also permits operation of the SD360 under control of the PDP-11/44.

The similar Spectral Dynamics SD350 single-channel spectrum analyzer lacks the multiple functions and transient capture features of the SD360, but has the capability to compute magnitude spectra on a wider variety of data ensembles (from 128 to 2048 points). The SD350 is also equipped with a real-time "waterfall" display for viewing

Table 3.1

Aircraft Acoustic Noise Recordings in
Speech Processing Facility Data Base
(Not Including Wordlist Recordings)

Aircraft (and position)	Recording Date and Source		Ref.	Absolute Noise Level
E-4B (battle staff)	1982	Ketron	3.7	88 dB (C)
E-4B (briefing rm.)	"		"	83 dB (C)
E-4B (NCA comp.)	"		"	78 dB (C)
EC-135 (radio oper.)	"		"	89 dB (C)
EC-135 (battle staff)	"		"	89 dB (C)
E-3A (console 4)	1979	RADC/EEV	3.9	86 dB (C)
E-3A (console 10)	"		"	86 dB (C)
E-3A (console 13)	"		"	87 dB (C)
E-3A (console 25)	"		"	86 dB (C)
E-3A (console 30)	"		"	86 dB (C)
EC-130 (ABCCC)	1984	RADC/EEV	-	102 dB (C)
EC-130 (Seat 1)	1984	RADC/EEV	-	102 dB (C)
HC-130	1984	RADC/EEV	-	95 dB (C)
P-3C	1978	Ketron	3.10	105 dB (C)
HH-53	1984	RADC/EEV	-	113 dB (C)
F-15	1979	AMRL	-	105-114 dB (C)

Table 3.1
Acoustic Noise Recordings

time-varying spectral characteristics. Finally, the SD350 has an IEEE bus interface that permits the PDP-11/44 to control, and exchange data with, the SD350. Further details of the SD350 are provided in Chapter 6 of this report and Ref. 3.4.

This study is based primarily on spectral estimation, and so is subject to the limitations (cf. Ref. 3.5) that apply to any spectral estimation approach. The SD350 and SD360 analyzers use the periodogram method of spectral estimation, with a Kaiser-Bessel window, and are capable of averaging periodogram magnitudes over time to reduce estimate variance. The Kaiser-Bessel window achieves excellent sidelobe suppression at the expense of a tolerably small loss of analysis resolution.

For signal analysis that does not fit into the framework provided by the SD350/SD360 functions, it is necessary to use the general-purpose capability of the PDP-11/44 system. Using a general-purpose system entails a loss in processing speed (which can be made up in part by use of the MAP-300 and FPS-120B array processors), but a gain in flexibility. A good example of this tradeoff presents itself if we wish to analyze the variation in acoustic noise on a fine time scale comparable to the 22.5 ms frame length typical in narrowband speech communication. Although the SD360 is capable of computing a spectrum in real time (for the 5-kHz audio bandwidth of most interest in speech processing), it cannot transmit its results to the PDP-11/44 in real time; in fact it takes on the order of 1 sec for the SD360 to send one spectrum to the PDP-11/44. To take a concrete example, if we wished to perform a 5-kHz analysis of a noise signal with a sliding series of 1024-point short-term discrete Fourier transforms with 50% overlap (to generate a transform every 50 ms), the SD360 standing alone would be able to do the computations and produce a real-time display on its CRT. But if we wanted to send the results to the PDP-11/44 for storage, analysis, and plotting, we would have to be content with throwing away 19 out of every 20 transforms while the SD360 sent its data to the PDP-11/44.

For analyses beyond the limits of the real-time spectral analyzers, the main tools have been: (i) the ARCON-developed digital speech data base, with its real-time digital sampling software MAPIN; (ii) the Interactive Laboratory System (ILS) software package; (iii) the IEEE Digital Signal Processing software package (Ref. 3.6); and (iv) special analysis programs written for specific purposes, using the capabilities of the digital speech data base and ILS, and in some cases using the the MAP-300 and FPS-120B array processors.

Calibration- - In this study we have analyzed noise power in terms of relative levels across the frequency domain. Thus our plots in the following section show relative levels, and it would not be meaningful to compare absolute acoustic noise levels in two aircraft by overlaying the plots we give here. In Section 3.6 we discuss how absolute levels could be obtained from the same data, without any new measurements.

3.4 EXAMPLES OF SPECIFIC AIRCRAFT

In this section we discuss some characteristic properties of acoustic noise samples taken from the RADC Speech Processing Laboratory's acoustic noise data base. These examples represent acoustic noise power spectral estimates with analysis ranges extending up to 8 kHz, generally averaged in magnitude over a period on the order of 10 seconds.

3.4.1 Acoustic Noise In The E-4B

The RADC Speech Processing Facility data base includes recordings made in June 1982 aboard the E-4B Advanced Airborne Command Post. The E-4B is based on the commercial Boeing 747 airframe and is the successor to the EC-135 as a strategic command and control platform. These recordings were made by Ketron Corp. (Refs. 3.7 and 3.8) in three areas of the E-4B: the battle staff work area, near the middle of the aircraft; the briefing room, just forward of the battle staff area; and the National Command Authority (NCA) compartment. The locations of these areas are shown in Figure 3.2.

Fig. 3.2 E-4B Approximate Recording Locations

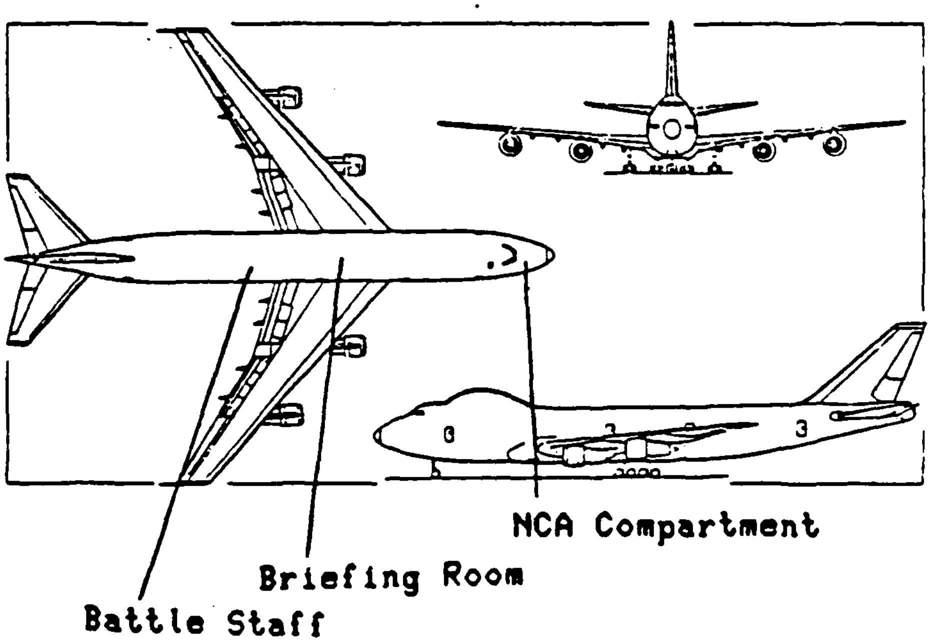
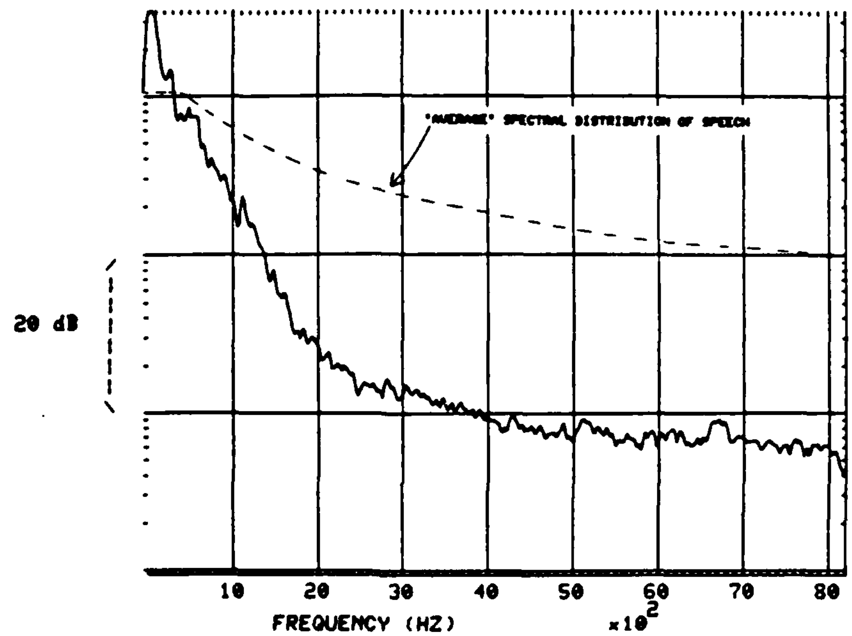


Figure 3.3 shows a typical noise power spectrum from the battle staff area, overlaid with the spectral shape of "average" speech. In general, acoustic noise above 700 Hz appears to be very well controlled in this aircraft. The most significant aspect of this noise environment is the low-frequency noise between 100 and 700 Hz. When noise from these recordings was analyzed in the frequency domain and averaged over 10-sec periods, the noise spectra showed:

1. fairly low absolute noise levels, 88 dB (C) in the battle staff area, 83 dB (C) in the briefing room, and only 78 dB (C) in the NCA compartment;
2. a "noise floor" characteristic of large aircraft with a peak near 100 Hz and a general slope of -10 dB per octave;
3. no strong isolated sinusoidal components;
4. only slow and small variation in the noise spectrum from one 10-sec period to the next.

Fig. 3.3 E-4B Battle Staff Area Acoustic Noise



In Figure 3.4 we can see evidence of the consistent spectral shape of the background noise in this aircraft; the plot shows 50 successive 10-sec average noise spectra from the NCA compartment, superimposed.

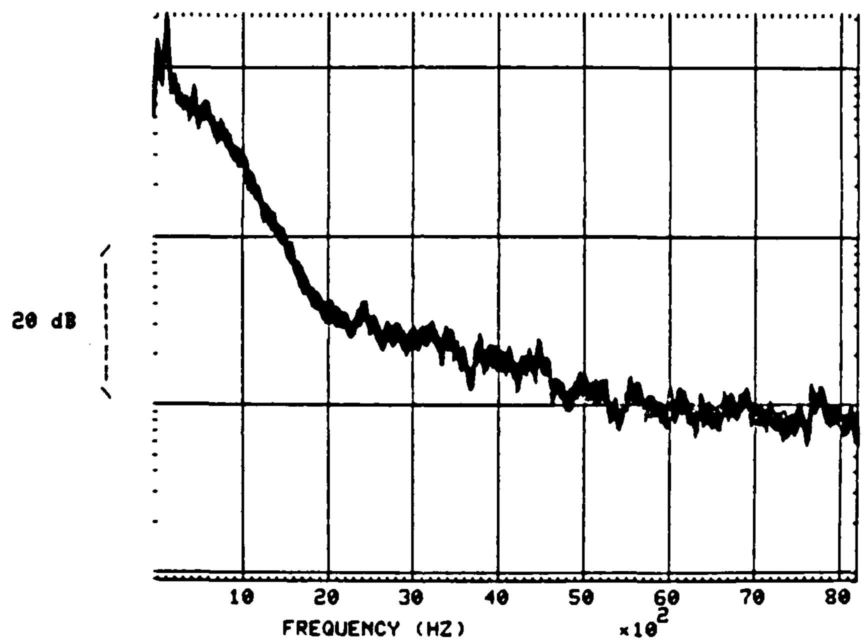
3.4.2 Acoustic Noise In The EC-135

The EC-135 is a modified version of the KC-135 tanker. As such it is similar to the commercial Boeing 707. Although the EC-135 is equipped for in-flight refueling of other aircraft, its primary function is command and control.

The Speech Laboratory's noise library includes recordings of ambient noise made on an EC-135 by Ketron Corp. in July 1982 (Refs. 3.7 and 3.8). These recordings were made at the Radio Operator's compartment and in the Battle Staff work area, as shown in Figure 3.5. Each recording was made with two microphones 52 inches apart, and each is about 10 minutes long. Analyses of these recordings show noise spectra with the following general characteristics:

1. an overall level of 89 dB (C) at both positions;
2. a "noise floor" characteristic of large aircraft with a general slope of about -10 dB per octave;
3. (at the Radio Operator position) strong discrete components at the harmonics of 1170 Hz;
4. (in the Battle Staff area) variations in the pattern of noise at frequencies above 2700 Hz, particularly between 2700 and 4000 Hz, but also extending upward to frequencies outside the usual range of speech processing.

Fig. 3.4. E-4B NCA Compartment Acoustic Noise



Figures 3.6 and 3.7 show typical noise power spectra at the two positions, and a superimposed plot of the long-term average spectral distribution of speech. Figure 3.8 shows several noise spectra (10-sec averages) from the Battle Staff area in a "waterfall" plot, showing the variability in noise in the 2700 - 4000 Hz range. During the recording there were several changes in flight control configuration; these are apparent to the ear and are confirmed by annotations accompanying the noise recording. The changes in 2700 - 4000 Hz noise seem to accompany throttle changes. On the other hand, there is little variation from one 10-sec period to the next in the recording made at the Radio Operator position. It may be that the aircraft remained in a single flight control configuration during the entire recording, but this cannot be confirmed from the existing documentation.

Fig. 3.5. EC-135 Approximate Recording Locations

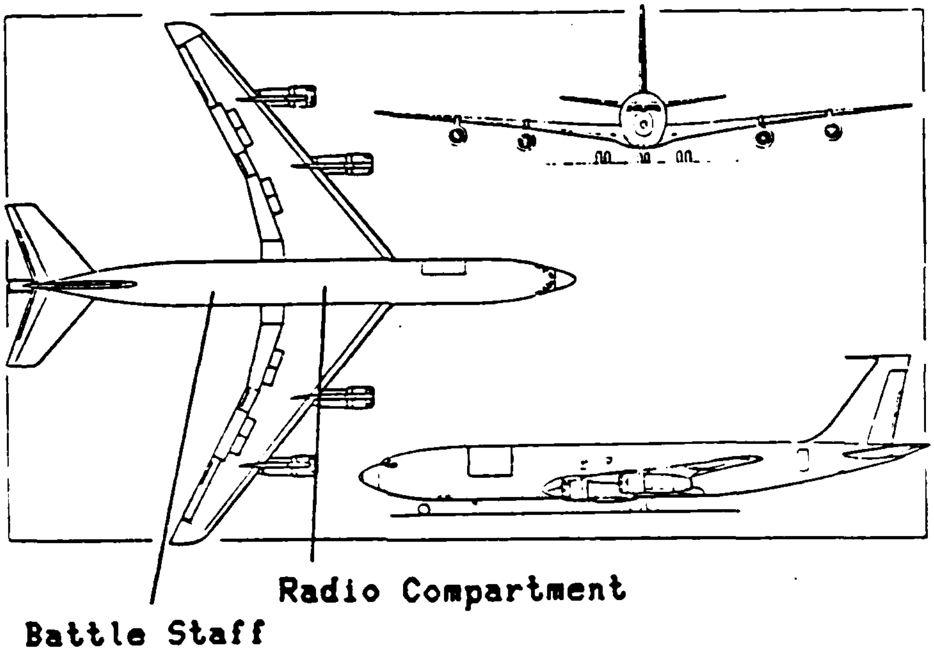


Fig. 3.6. EC-135 Battle Staff Area Acoustic Noise

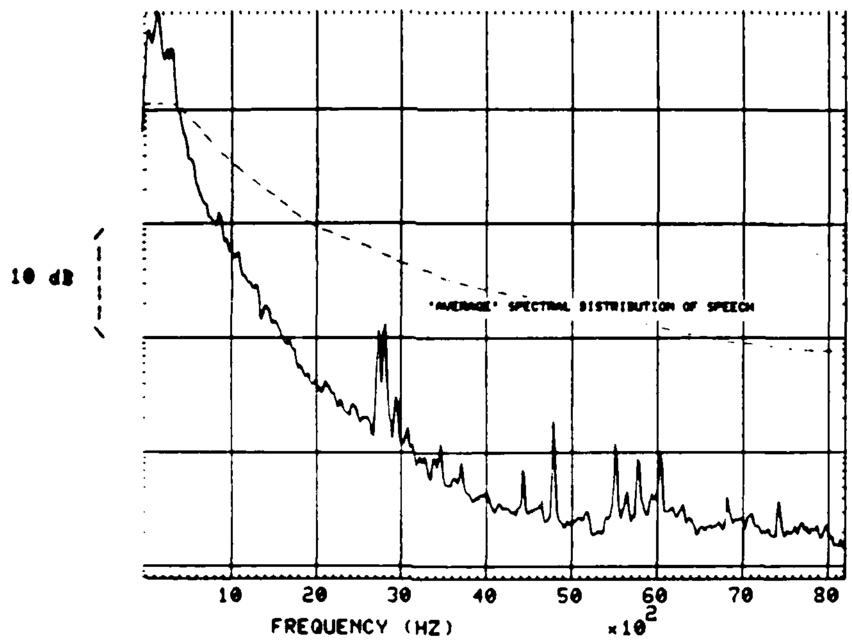


Fig. 3.7. EC-135 Radio Compartment Acoustic Noise

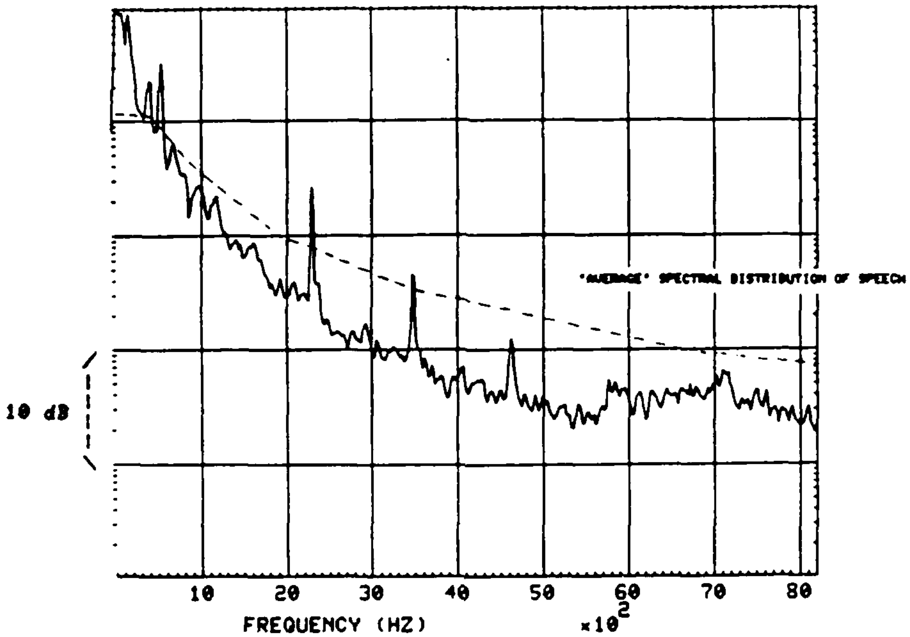
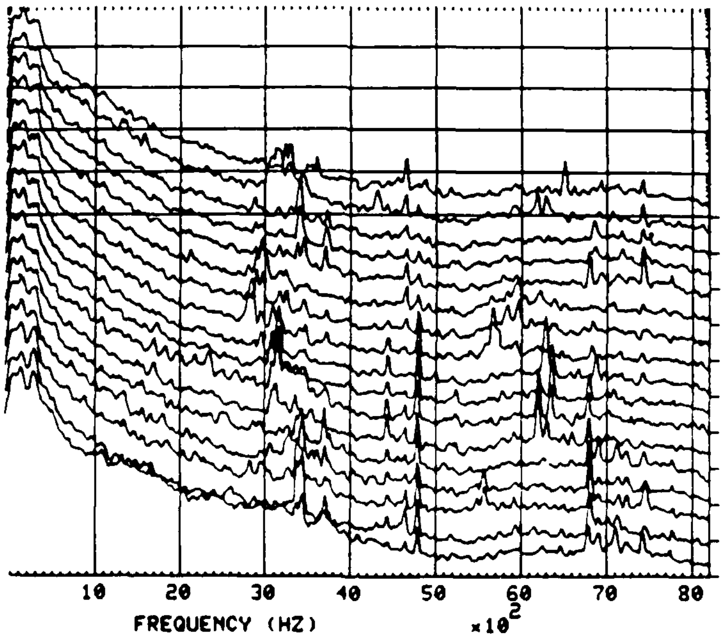


Fig. 3.8. EC-135: Variation of Noise in Battle Staff Area



3.4.3 Acoustic Noise In The E-3A

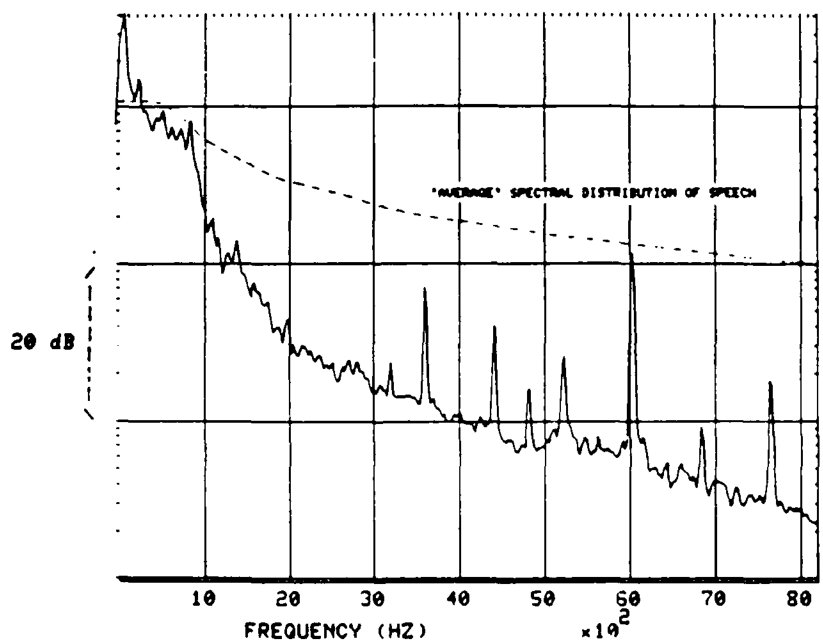
The E-3A (AWACS) carries a large radar and a crew of radar operators who track hostile targets and control fighter aircraft. The E-3A shares the same basic airframe used in the EC-135 and the commercial Boeing 707.

The Speech Laboratory's noise library includes recordings of ambient noise at several of the operators' consoles in an E-3A (Ref. 3.9). The recordings were made during a training mission while operators were present and speaking. In order to focus on the components of the acoustic noise due to the aircraft itself, the analysis was restricted to a few segments of the recordings in which speakers were not present in the immediate vicinity of the microphones. The analysis of these recordings showed noise spectra with the following general characteristics:

1. an overall noise level near 86 db (C) at all recording locations;
2. a "noise floor" characteristic of large aircraft with a general slope of about -12 dB per octave;
3. a number of significant discrete components, often including a family of discrete components spaced about 850 Hz apart and extending to 6000 Hz;
4. very strong discrete components at frequencies below 100 Hz.

Figure 3.9 shows an acoustic noise spectrum typical of those measured in the E-3A, with a superimposed plot of the long-term "average" spectral distribution of speech. The components 850 Hz apart do not appear equally prominent in all the measurements made on the E-3A.

Fig. 3.9. E-3A Console 13 Acoustic Noise



Figures 3.10 and 3.11 show in finer detail the spectrum of noise between 0 and 1200 Hz, as measured by two microphones at the E-3A's operator console #13. Each of these figures is a superimposed plot of 6 successive 10-sec average spectra measured on one of the two channels at this console. In most respects the two plots are similar, but just as in the EC-130 measurements mentioned later in this report, there is a marked difference in the spectra below 100 Hz.

Fig. 3.10. E-3A Console 13 Acoustic Noise 0-1200 Hz, Ch. 1

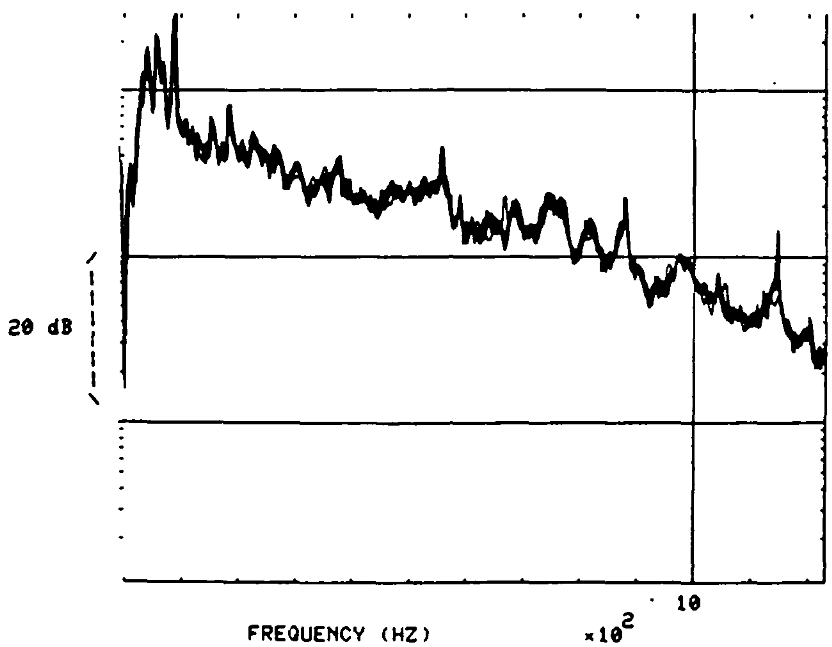
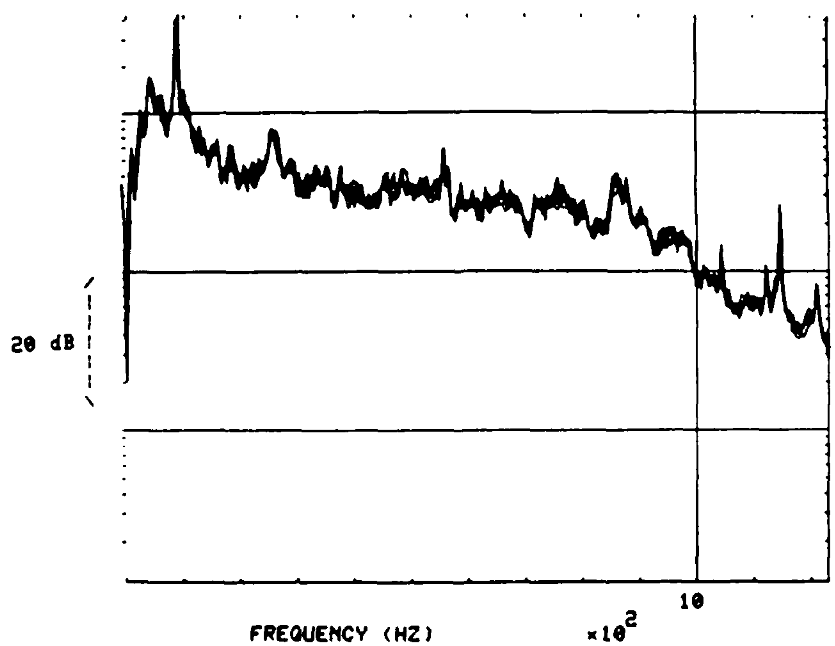


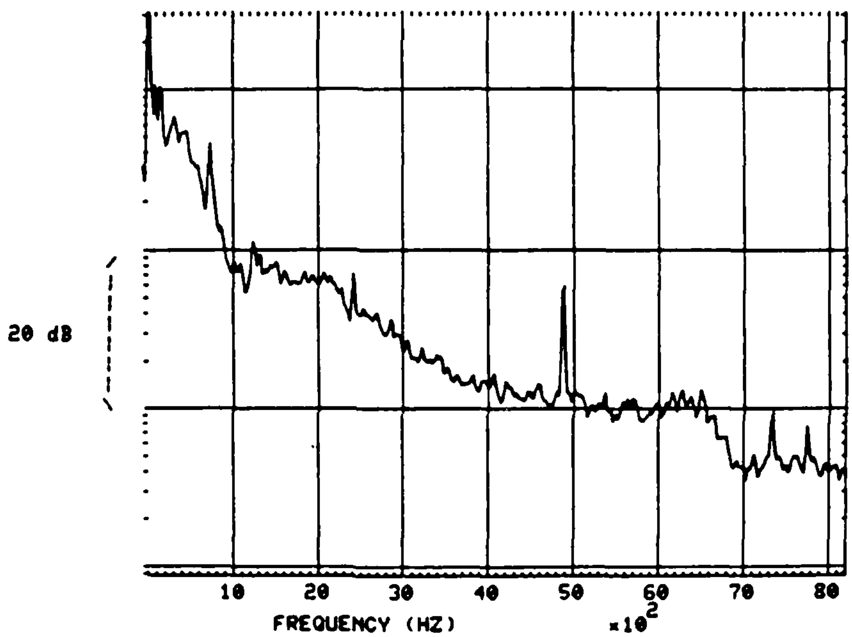
Fig. 3.11. E-3A Console 13 Acoustic Noise 0-1200 Hz, Ch. 2



3.4.4 Acoustic Noise In The EC-130 And HC-130

The EC-130 is a multi-engine turboprop aircraft, a version of the C-130 equipped for the command and control function. The HC-130 is a search and rescue variant of the same basic airframe. Among the EC-130 noise recordings made in 1984 by RADC/EEV personnel, there is a notable variation in spectral shapes between noise recorded at one time or location and another, including significant variations between recordings made simultaneously with two microphones a few feet apart. However, a representative noise power spectrum is given in Figure 3.12.

Fig. 3.12. EC-130 Representative Acoustic Noise



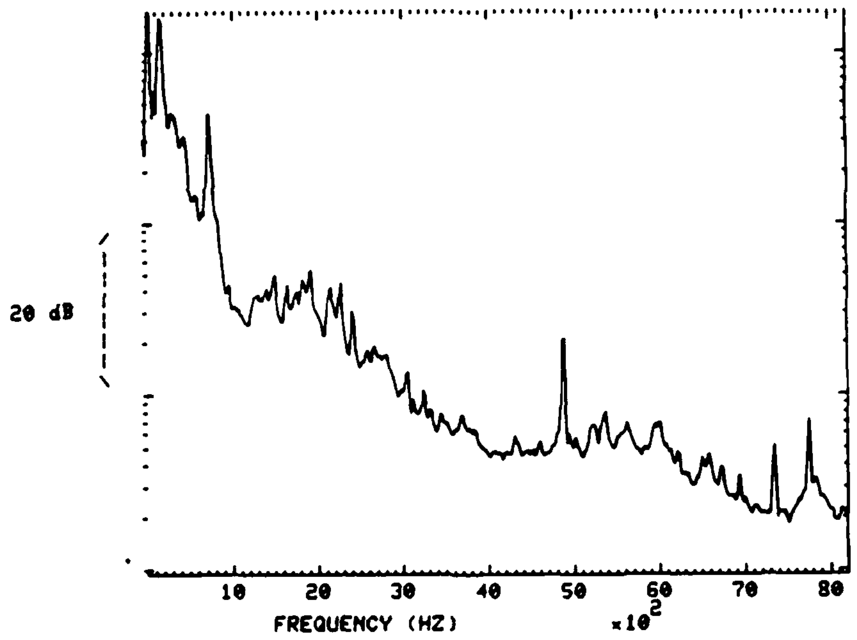
The acoustic noise environments measured in the EC-130 appear generally to be characterized by:

1. an overall level of 102 dB (C) at both locations in the EC-130;
2. a "noise floor" characteristic of large aircraft with a general slope about -8 dB per octave;
3. three strong discrete components between 70 and 210 Hz;
4. a discrete component sometimes appearing between 750 and 800 Hz;
5. occasional discrete components 6-10 dB above the "floor," between 1000 and 5000 Hz;
6. a strong discrete component near 4900 Hz.

Discrete Components - The acoustic noise power spectra measured in the EC-130 are dominated by three features at 70-210 Hz, 750-800 Hz, and 4900 Hz. The 750-800 Hz feature is not always present. In some measurements, additional discrete noise appears between 2500 and 4900 Hz.

The discrete components with the highest energy appear at 70, 140, and 210 Hz. We hypothesize that these frequencies represent rotation rates associated with the turbines and gear trains in the turboprop engines. The relative level of the three spectral lines varies significantly from one noise recording to another, and from one microphone position to another within single two-channel recordings; but all recordings show at least the 70 Hz component. An example of this variability is shown in Figures 3.13 and 3.14, which show noise power measured by two microphones 30 inches apart at the Airborne Communications, Command, and Control (ABCCC) position. Microphone 1 measured roughly equal peaks at 70 Hz and 210 Hz. At the same time, and consistently over a period of minutes, Microphone 2 measured a peak 10 dB lower at 210 Hz than at 70 Hz.

Fig. 3.13. EC-130 Acoustic Noise, Microphone 1



Another noise peak appears in the 750-800 Hz interval, but only in recordings made at the ABCCC position (Figures 3.13 and 3.14). This peak may be harmonically related to peaks near 2500, 3300, 4100, and 4900 Hz —spaced approximately 800 Hz apart—measured elsewhere in the aircraft (Figure 3.15). A very strong and consistent discrete component appears at 4900 Hz in all the recordings made on the EC-130, as can be seen in Figures 3.12 through 3.15.

Fig. 3.14. EC-130 Acoustic Noise, Microphone 2

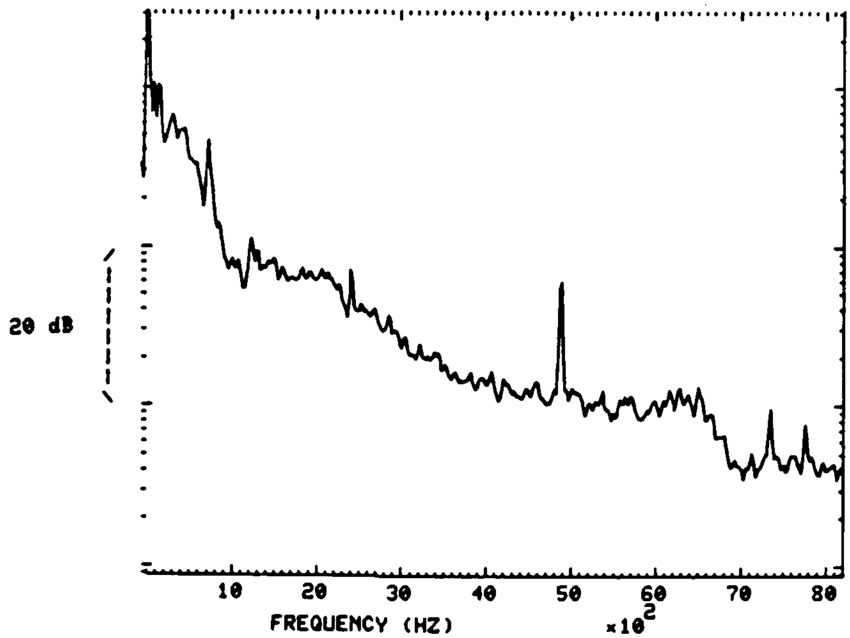
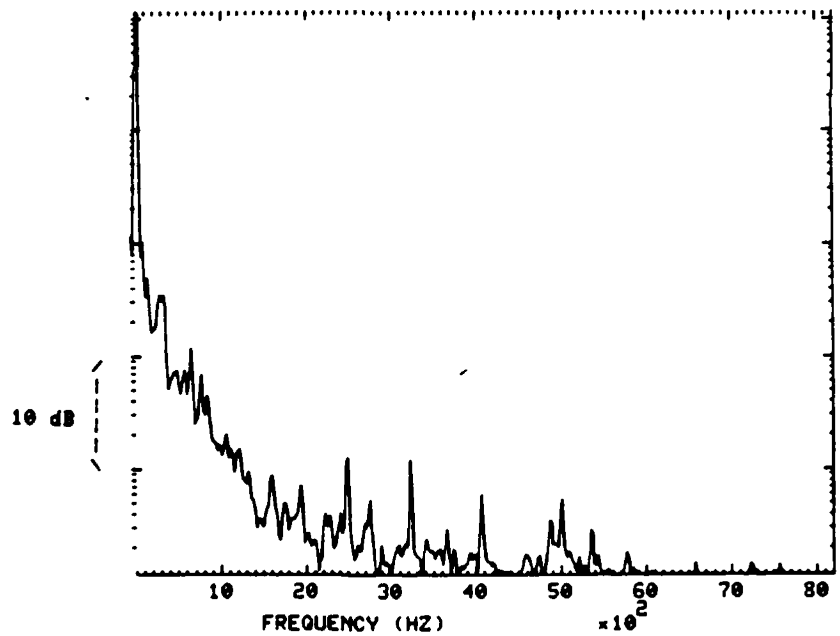
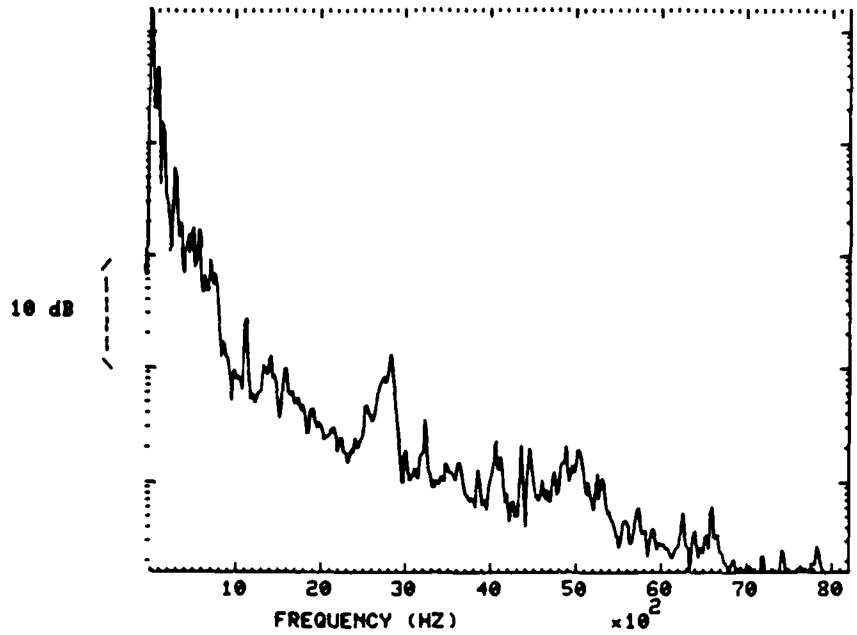


Fig. 3.15. EC-130 Acoustic Noise, Seat #1 Position



Broad Resonances - Broad peaks appear in some of the noise spectra measured in the EC-130. One appears around the 750-800 Hz peak already mentioned, and could represent either a rapidly varying discrete component or a stationary noise source with a narrow bandwidth less than 200 Hz. Another broad peak appears in some, but not all, measurements made at the Seat #1 position on the aircraft, between 2500 and 3000 Hz (Figure 3.16).

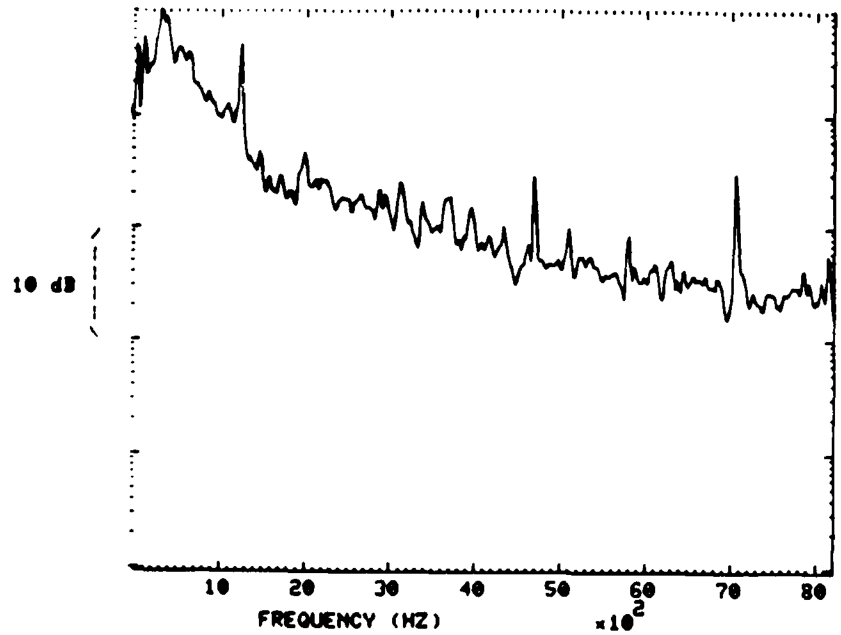
Fig. 3.16. EC-130 Acoustic Noise with 2500-3000 Hz Peak



Comparison With HC-130 Noise Environment - In comparison with the EC-130, noise spectra measured over a short period in one location in the similar HC-130 aircraft show (Figure 3.17):

1. a lower overall level, 95 dB (C);
2. the same low-frequency peaks at 70 and 140 Hz;
3. discrete components at 1200, 4700, and 7100 Hz.

Fig. 3.17. HC-130 Acoustic Noise



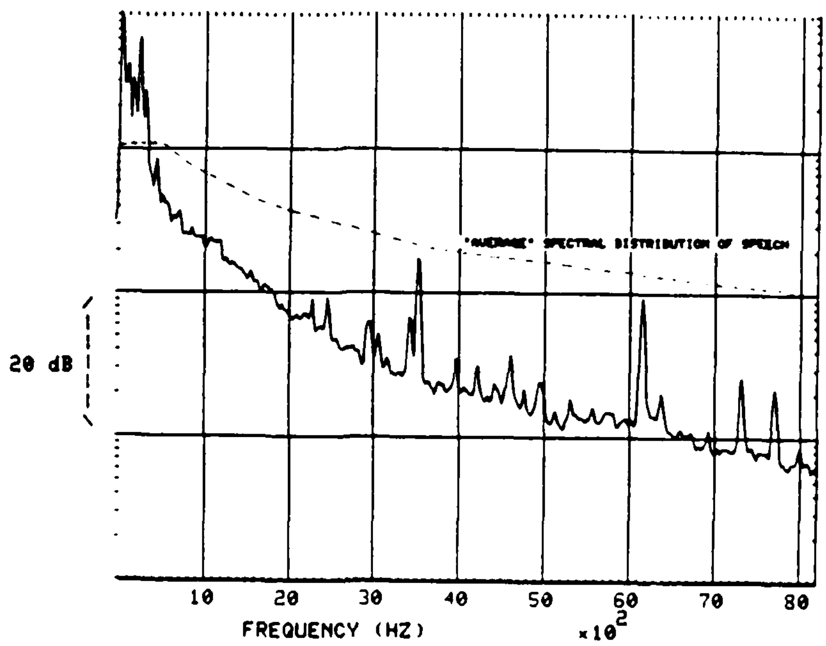
3.4.5 Acoustic Noise In The P-3C

The P-3C is a long-range anti-submarine patrol aircraft, developed from the commercial Lockheed Electra and used by the U. S. Navy. The Speech Laboratory's noise library includes a short recording (about 40 sec) of noise in a P-3C in flight. The recording was made by Ketron Corp. for a 1978 study (Ref. 3.10) in support of ANDVT development. Analysis of this brief recording shows noise spectra with the following general characteristics:

1. an overall level of 105 dB (C);
2. a "noise floor" characteristic of large aircraft with a general slope of about -8 dB per octave;
3. a concentration of noise power below 500 Hz;
4. very strong discrete components near 3600 and 6100 Hz;
5. a number of less powerful discrete components all across the range from 3200 Hz to 8000 Hz (the maximum frequency for our analysis).

When the 40-sec recording was broken into separate 10-sec intervals for analysis, there was little variation between the measured noise spectra. A typical spectrum is shown in Figure 3.18, with a superimposed plot of the long-term "average" spectral shape of speech.

Fig. 3.18. P-3C Acoustic Noise



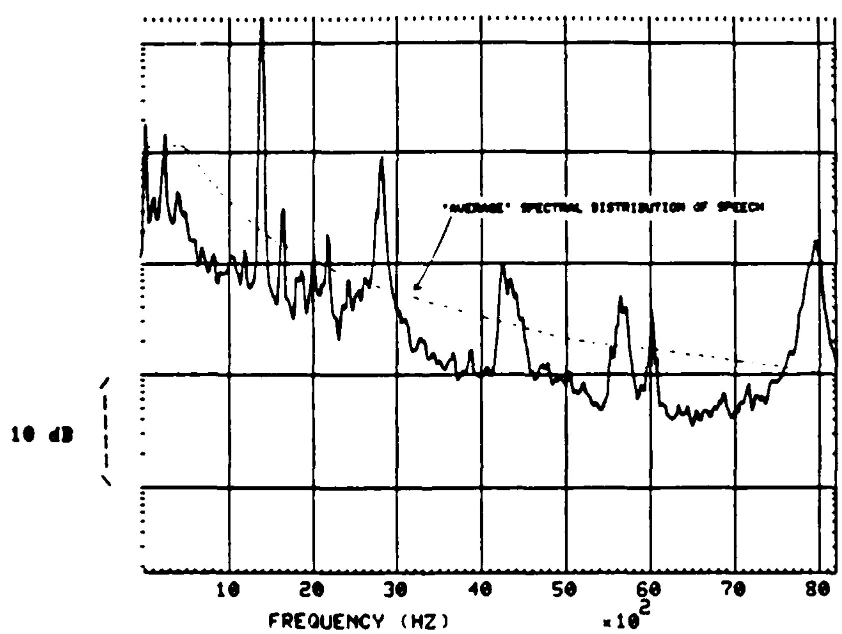
3.4.6 Acoustic Noise In The HH-53 Helicopter

Helicopter noise has presented severe problems for narrowband speech communication systems. In this section we will see some indications of why helicopter noise is so taxing for parametric speech coding in particular.

The RADC Speech Processing Facility data base includes a single helicopter noise recording, made in 1984 by RADC/EEV personnel aboard an HH-53 helicopter in flight. The HH-53 is a search and rescue helicopter with main and tail rotors powered by a turbine engine. The recording is about 8 min in length and was made with two microphones. When this recording is analyzed in the frequency domain, a 10-sec average spectrum like that shown in Figure 3.19 is typical. The most striking features of the acoustic noise background are:

1. a high overall noise level, 113 dB (C);
2. a higher proportion of energy at middle and high frequencies than is typical of larger aircraft;
3. strong discrete sinusoidal components at frequencies in the middle of the speech frequency band; and
4. rapid variation of noise related to the main rotor's rotation, especially during periods of "blade slap".

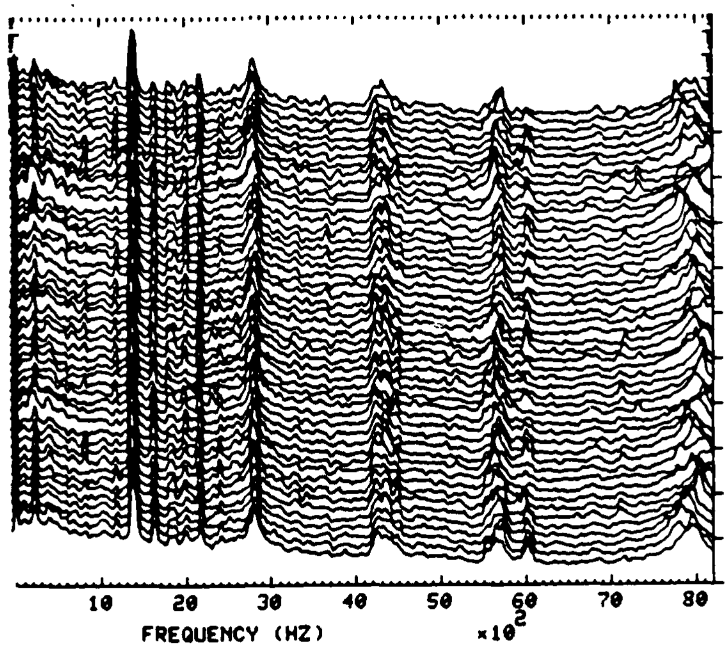
Fig. 3.19. HH-53 Acoustic Noise



As Figure 3.19 shows, the measured noise power levels in the HH-53 do not fall off as steeply as "average" speech does with increasing frequency. Therefore we would expect speech analyzers to have increased difficulty with higher formants. The concentration of noise power above 1000 Hz is especially troublesome because noise-canceling microphones do not have much effect at such high frequencies.

Figure 3.20 shows HH-53 noise power spectra as a function of time. Each spectrum is an average over approximately 10 sec, and so the graph shows only the longer-term variation pattern. A number of apparently sinusoidal components can be seen appearing and disappearing over the course of the analysis period; this variation may be linked with clearly audible periods of "blade slap" during the recording.

Fig. 3.20. HH-53: Successive 10-sec Average Spectra



The strong sinusoidal component observed at 1460 Hz and the broader 2920 Hz (= 1460 + 1460) peak pose particular problems for speech processing. They appear in the range of frequencies normally occupied by the perceptually important second and third vocal-tract formants, and would tend to affect processing by displacing these formants.

3.4.7 Acoustic Noise In The F-15

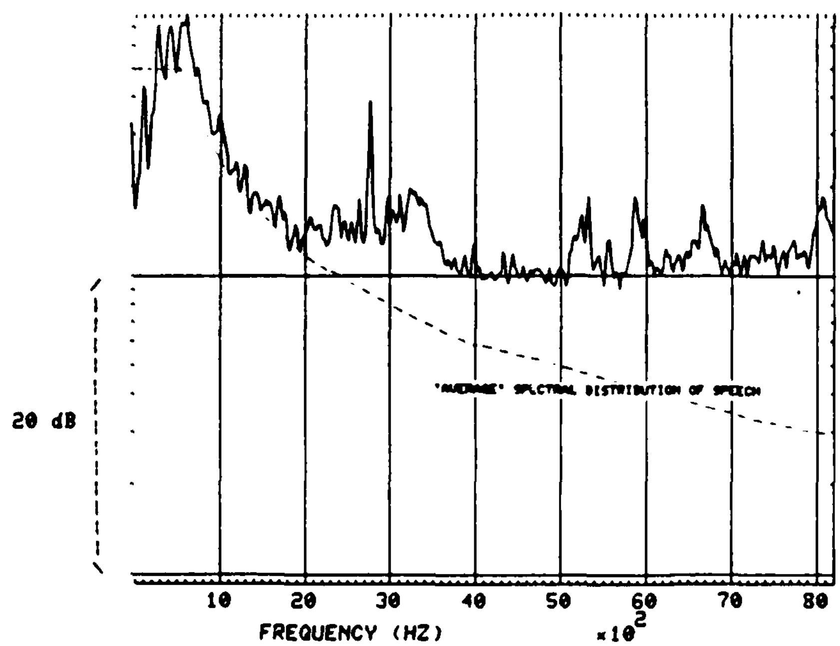
The F-15 is a twin-engine single-seat air-superiority fighter. The pilot of the F-15 works in higher noise levels than do the crew of larger aircraft, but is to some extent shielded from the effects of this noise by his helmet and oxygen mask. The location of the pilot's microphone (inside the oxygen mask) effects a very considerable reduction in the acoustic noise picked up by the microphone.

The Speech Laboratory's acoustic noise data base includes a recording made aboard an F-15 at Wright-Patterson Air Force Base in 1976. Unfortunately, there is no absolute calibration for this tape and so we can say nothing about the absolute noise levels. However, another study (Ref. 3.11) has found noise levels of 105-114 dB (C) in the F-15 cockpit during flight.

The 1976 recording includes short segments of noise measured during level flight at 6000, 25000, and 40000 ft; during a climb from 10000 ft to 25000 ft; and during a climb from 25000 to 40000 ft. In addition there are segments of noise measured while the aircraft was on the ground. Figure 3.21 shows a noise spectrum averaged over a 10-sec period during subsonic level flight at 25000 ft. Some common features of the F-15 noise environment appear in this sample spectrum:

1. a broad "hump" in the spectrum at frequencies below about 1200 Hz;
2. more high-frequency noise than in larger aircraft;
3. a strong discrete noise component near 3000 Hz;
4. broader peaks in the range 3000 - 8000 Hz.

Fig. 3.21. F-15 25000 ft 0.95M level flight



The long-term "average" spectral distribution of speech is shown in the dotted line for comparison. In larger aircraft the noise level drops off more or less steadily with increasing frequency, so that the weaker high-frequency portions of the speech spectrum are competing with the weakest part of the noise spectrum. However, in this aircraft there is no "rolloff" in noise power above 2000 Hz; in fact, above about 4000 Hz the noise level actually tends to increase as the frequency increases.

Figure 3.22 shows a noise spectrum at the same altitude in supersonic flight (Mach 1.3). The dominant discrete noise component is even more pronounced. It appears at a higher frequency, probably because of a higher turbine rotation speed at the higher airspeed.

During climbs, there seem to be more pronounced discrete components in the noise. Figures 3.23 through 3.25 show noise spectra obtained as the F-15 was climbing through 15000, 20000, and 35000 ft. Overall broadband noise levels were generally lower during climbs than during level flight at the same airspeeds, but during a climb near 40000 ft at Mach 0.88 with full military power, the measured spectrum (Figure 3.26) has a discrete component even stronger than usual.

Fig. 3.22. F-15 25000 ft 1.35M level flight

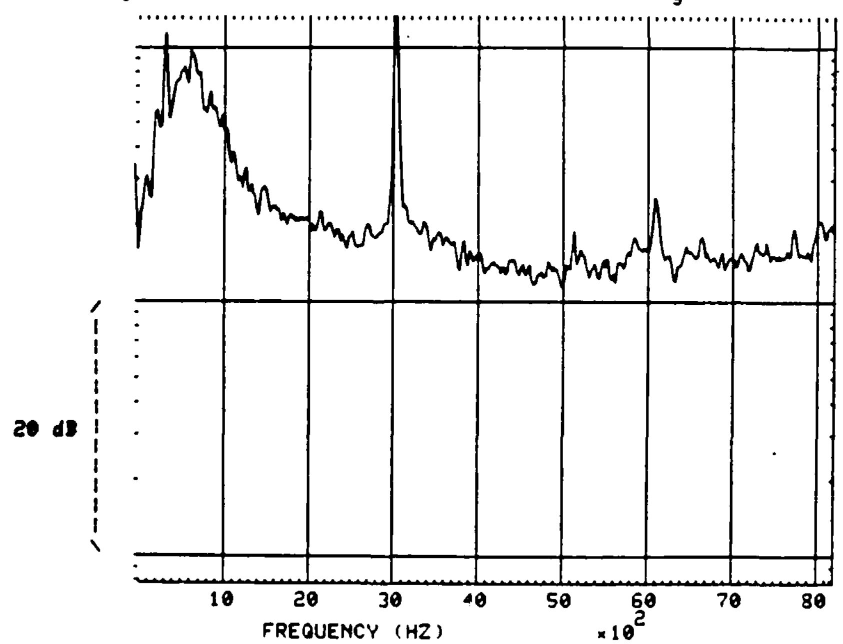


Fig. 3.23. F-15 15000 ft 0.92M climbing

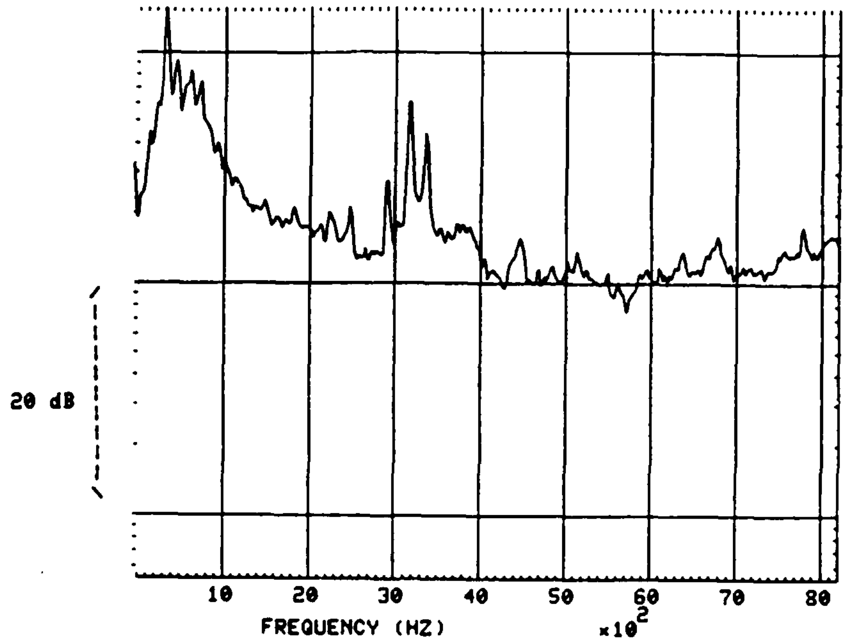


Fig. 3.24. F-15 20000 ft 0.88M climbing

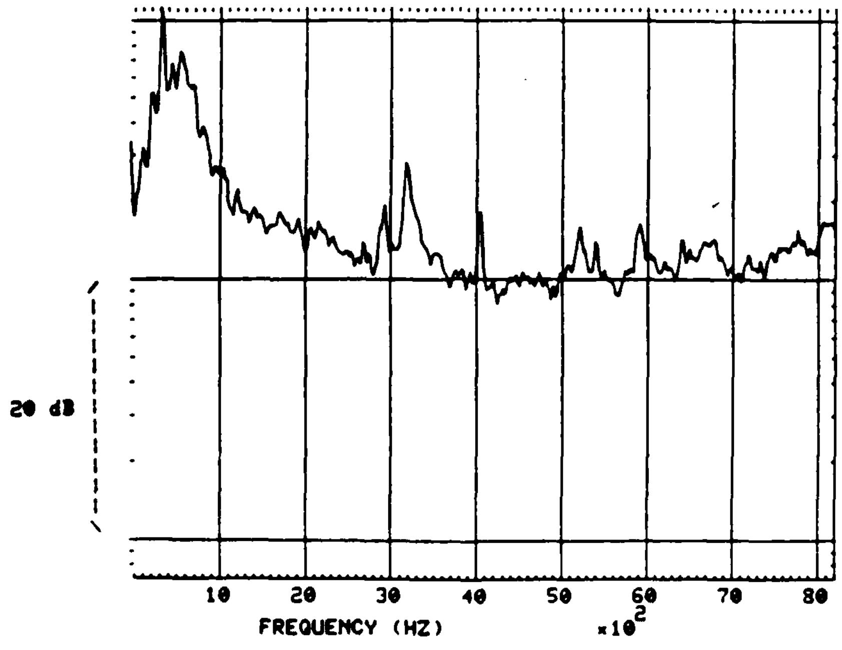


Fig. 3.25. F-15 40000 ft 0.90M climbing

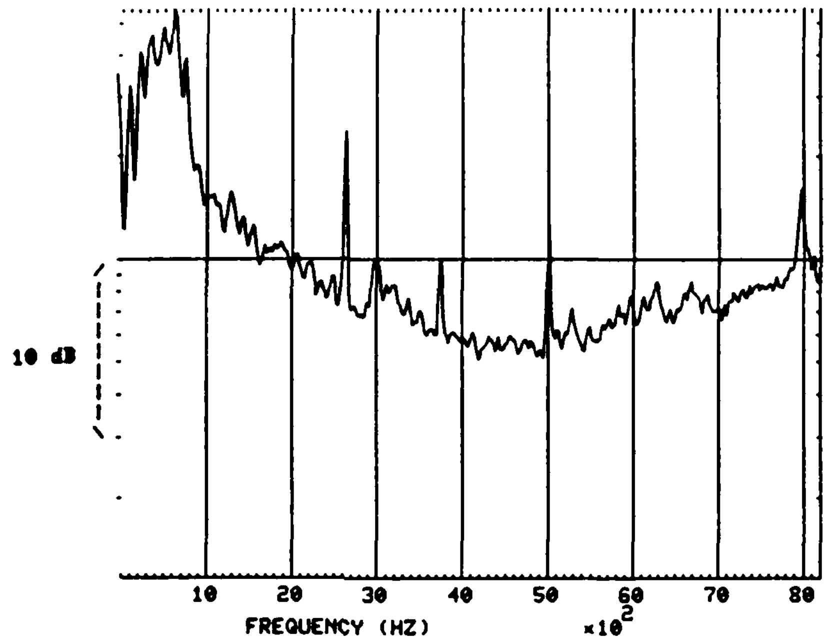
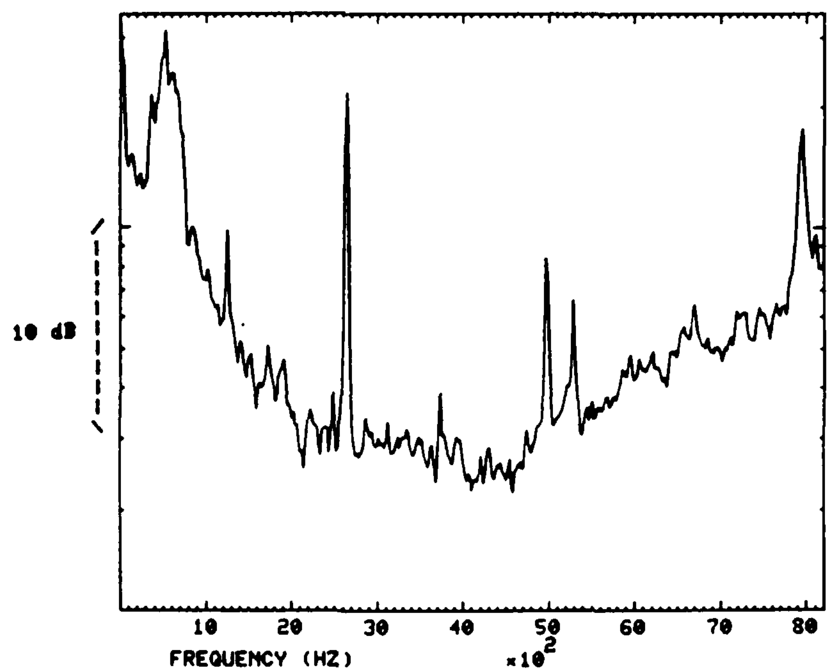


Fig. 3.26. F-15 40000 ft 0.88M climbing, mil pwr



3.5 RELATIONSHIP TO NOISE SUPPRESSION AND REDUCTION METHODS

In discussing approaches to the problem of processing noisy speech, it is useful to distinguish between preprocessing approaches (which modify the speech signal before it is presented to the analyzer) and analysis approaches (which use special analysis methods intended to be more robust in the presence of background noise). Pre-processing approaches have the advantage that they may be usable with more than one type of analyzer, or in situations where the choice of analyzer is dictated by other considerations such as interoperability. However, this is not to say that a pre-processing approach will be equally successful with all types of analyzers. Analysis approaches, on the other hand, can use techniques peculiar to one class of analyzer, such as the channel vocoder, or could conceivably involve a whole new class of analyzers, and could be advantageous in situations not otherwise constrained to a particular analysis method.

In a parametric analyzer (and therefore in any current low-bit-rate vocoder), background noise presents an inherent problem. This is of course because the background noise is not accounted for in the parametric model on which the analyzer depends. The unmodeled background noise is not analyzed correctly during the analysis, and it is not reproduced correctly during the synthesis.

Attempts to develop preprocessors for robust narrowband speech communication systems in the presence of acoustic background noise have concentrated on three avenues:

1. designing transducers to reject acoustic signals originating at a distance from the speaker;
2. noise cancellation (generally with adaptive filters) to monitor and remove background noise in the time domain; and
3. spectral subtraction to remove background noise in the frequency domain based on the estimated magnitude spectrum of the background noise.

All these approaches can be made to produce a speech output signal, and therefore can be used with any analyzer that expects speech input. However, the effectiveness of the preprocessor-analyzer tandem may depend on the interaction of the preprocessor and the analyzer. It appears to be much easier to improve the subjective quality of speech processed in background noise than to improve its intelligibility. In general, trouble can be expected if the pre-processor provides noise rejection at the expense of introducing distortion of the speech signal; an example of this type of problem is given below in our discussion of the spectral subtraction technique.

Transducers - The surest way to control the effects of background noise is to keep it from mixing with the speech signal in the first place. In a quiet environment, existing noise-cancelling microphones do not perform as well as standard microphones. However, if it is known that the background noise level will be high, noise-cancelling microphones are worth considering. The noise-cancelling microphones now in the field do significantly reduce background noise at low frequencies (Ref. 3.12). However, these microphones introduce spectral peaks that distort the shape of the speech spectral envelope.

The performance of parametric analyzers, in particular, will be degraded by these distortions. More recently, noise-cancelling microphones have been designed with an essentially flat response in the frequency range of interest for speech analysis, but even these microphones do not provide much noise reduction above 1000 Hz or so (Refs. 3.7 and 3.12). We can conclude that noise-cancelling microphones alone would not be very helpful in a helicopter noise environment like that shown in Figure 3.19, in which most of the noise energy is above 1000 Hz. Recently, there has been interest in combining the outputs of multiple microphones or accelerometers to produce a cleaner speech signal (Ref. 3.13). Although the RADC Speech Processing Laboratory is not equipped for transducer research and development per se, further developments in this area could have an impact on the performance of speech communication systems in acoustic noise.

Adaptive Noise Cancellation - This method (Ref. 3.14) can be regarded as a special case of multi-sensor processing. It requires, in addition to the primary "speech + noise" signal, a reference "noise" signal highly correlated with the noise in the primary channel. It is not in general assumed that the noise in the reference channel is identical to the additive noise in the primary channel, but instead a dynamically updated linear filter is applied to the reference channel to form an estimate of the noise present in the primary channel. The coefficients of this filter are dynamically adapted, either continuously or during "non-speech" frames, in order to minimize the energy in the noise-cancelled output. This computationally expensive technique has been tried by RADC/EEV in a helicopter noise environment (Ref. 3.15), but produced only a slight increase in intelligibility scores. In a variation of this technique, Kang and Everett (Ref. 3.12) have suggested placing the reference microphone quite close to the speaker, but just far enough away that it receives little speech power, and taking the reference output itself as the noise estimate, with no intervening adaptive filter.

Spectral Subtraction - This technique (surveyed in Ref. 3.1) can be either used as a pre-processing step, or (if the analyzer operates in the frequency domain) incorporated into the analyzer. Spectral subtraction does not require a second noise-only input, because it uses an estimate of the long-term noise spectral density instead of trying to estimate the noise signal itself. The estimated noise magnitude spectrum is updated during periods of no speech, and therefore it is necessary to have a speech present/absent decision algorithm. During speech, the speech + noise signal is transformed to the frequency domain. The magnitude spectrum is then modified at each frequency, based on the magnitude of the noise estimate; several different rules have been proposed to decide precisely what the revised magnitude should be as a function of the speech + noise magnitude and the noise magnitude (Refs. 3.16 - 3.18). Generally the phase of the speech + noise signal is left intact (although other options have been suggested), and the resulting complex spectrum is then transformed back into a time signal for input to the analysis system. In the case of a channel vocoder or other analyzer operating in the frequency domain, this final transformation would not be necessary. Spectral subtraction is also convenient to add to an analyzer hardware that is powerful enough to transform quickly between the time and frequency domain, even if the analyzer is time-based.

The spectral subtraction technique has been used for noise suppression with little success in improving intelligibility as measured by DRT scores (Ref. 3.19). Since spectral subtraction depends on the estimation of noise spectra during non-speech periods, it is particularly susceptible to variations in the noise spectrum that may occur on a time scale smaller than several seconds. Some of the noise measurements detailed in Section 3.4 above suggest the presence of such variations.

Even if the noise is stationary in a statistical sense, spectral subtraction techniques are limited further because of the random variation of even a statistically stationary noise source. For example, the magnitude of Gaussian noise at a particular frequency "bin" will vary about its long-term average magnitude with a standard deviation equal to the long-term average magnitude itself. This statistical variation is blamed for the "musical tones" often noticed in speech preprocessed by spectral subtraction.

For parametric coding, if preprocessing distorts the speech signal while trying to remove the noise, then the problem of unmodeled noise referred to above becomes an even more difficult problem of unmodeled noise and unmodeled distortion. In particular, if the "local" signal/noise ratio in a narrow frequency band is low, a spectral subtraction technique is likely to remove both noise and signal. For example, what would spectral subtraction do with the helicopter noise spectrum *shown in Figure 3.19, which has peaks 1460 Hz and 2920 Hz?* Removal of these peaks would also remove formant information present in this important range of frequencies, perhaps improving the subjective quality of the preprocessed speech, or even the subjective quality of the vocoded and synthesized signal, without improving intelligibility.

3.6 CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

3.6.1 Time Variation Of Acoustic Noise

The measurements presented in Section 3.4 show, in several cases, significant variation in acoustic noise between one 10-second average and a later 10-second average. But it appears that much of this variation is due to changes in the control configuration of the aircraft. Such changes are to be expected during operations in tactical aircraft, but may be less important in aircraft that operate in a communications, command and control role. Within the recordings studied here, the inherent acoustic noise environments of the communications, command and control aircraft were quite stable.

While this study has only investigated noise averaged over 10-sec periods, we recommend further investigation of the variation of acoustic noise over shorter periods, comparable to the 20-30 ms length of a typical "frame" of speech analysis.

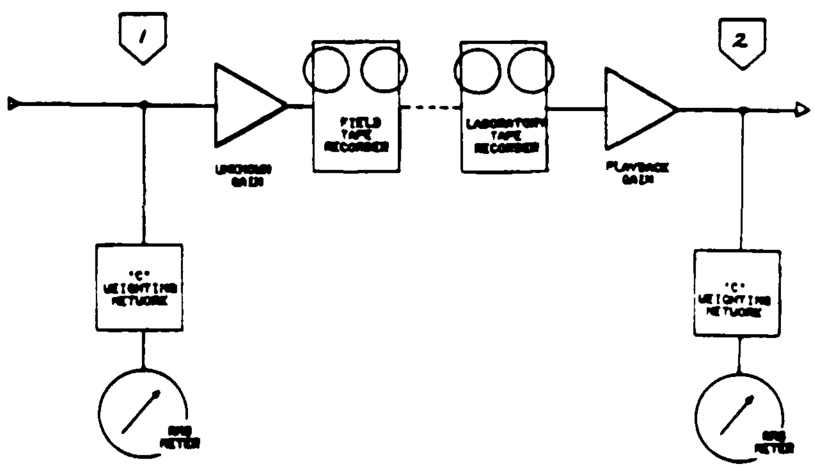
3.6.2 Spatial Variation Of Acoustic Noise

This study has documented differences in acoustic noise from one compartment to another in the same aircraft, and also substantial and repeatable differences between noise measured by two microphones as little as 30 inches apart. These differences show that we should not expect high correlation between acoustic noise at two locations near one another, and further imply that detailed analysis of particular acoustic noise spectra is of limited predictive value.

3.6.3 Absolute Calibration Of Noise Power Estimates

Noise tapes in the Speech Laboratory's collection, coming as they do from various sources, use different microphones, attenuators, and tape recorders; but in most cases they are accompanied by absolute sound level measurements with A, B, and/or C weighting. In order to obtain information about the absolute sound pressure levels of ambient noise, or to compare the relative levels of two noise recordings made under different circumstances, it is necessary to compensate for the effects of microphone sensitivity, gain in the record and playback amplifiers, and the record and playback heads themselves—all of which may differ from one noise recording to another. The recording and playback systems have an essentially flat response over the audio frequency range, and therefore will introduce a frequency-independent gain, which must be measured and compensated for. What is needed is an "overall measurement system gain" (say, in volts squared per 20 micropascals) that relates squared voltage at the input to the A/D converters to sound pressure at the recording microphone. This factor can be obtained by measuring the mean square signal at the A/D input during playback, and comparing the (acoustic) sound level measured during recording, as shown in Figure 3.27. This method of calibration requires that the same weighting network (A, B, or C) be used for both measurements.

Fig. 3.27. Absolute Calibration of Field Recordings



3.6.4 Classification Of Acoustic Noise Environments

In terms of the frequency-domain characteristics of their acoustic noise environments, the aircraft studied here divide naturally into two groups. In the first group are large aircraft with wing-mounted engines. Aboard these aircraft, the bulk of the acoustic noise power is concentrated at frequencies less than 1000 Hz. Above 1000 Hz, the noise power drops off at 6-12 dB per octave, compared to the decline of 6 dB per octave in typical speech signals. As we have pointed out, such a shape is desirable both in terms of reduced competition with higher-formant information in speech and in terms of susceptibility to noise-cancelling microphones.

The second group consists of "other" aircraft. Out of the wide range of aircraft not falling into the first group, our data base covers only the HH-53 helicopter and the F-15 fighter. Although these two aircraft have little in common in terms of mission, aerodynamics, or propulsion, in both of these aircraft there is substantial noise power distributed all across the frequency range studied, and even higher. Noise-cancelling microphones are of little help with this high-frequency noise. Moreover, this study has found strong discrete components varying in frequency, which would be expected to cause severe problems for spectral subtraction processing techniques.

In the future, this classification should be expanded by comparing spectral shapes measured from the Speech Processing Facility's noise data base with third-octave analyses from the Air Force Aerospace Medical Research Laboratory acoustic noise data base. The objective of this comparison would be to determine whether there is evidence that aircraft not represented in the Facility's noise data base have noise environments significantly outside the range of those already represented. This investigation could be confined to aircraft for which there is a significant interest in secure voice communication. In the event that this comparison showed a need for further data, field recording efforts might be in order.

At the same time, we recommend collection of a set of digital noise records, complementing the existing analog noise recordings in the Speech Processing Facility's noise data base, and representing the ᶠull range of acoustic noise environments represented in that data se. These noise records could be incorporated in the existing digital speech data base.

Finally, we recommend research directed towards formulation of design parameters, as a function of the specific aircraft, for future speech compression algorithms that are intended to perform in Air Force noise environments. A major difficulty in this undertaking is the multiplicity of compression methods in use. In order to obtain specific design parameters, it may be advisable to deal broadly with two classes, waveform reconstruction methods and parametric modeling methods. In addition it may be necessary to limit the parametric class to all-pole modeling. This research could provide direction in the choice of error metrics for use with the canonical coordinate speech compression methods described in Chapter 2 of this report.

CHAPTER 4

RADC/EEV SPEECH PROCESSING FACILITY COMPUTER SYSTEM

The computer system hardware at the RADC/EEV Speech Processing Facility has been upgraded during the contract period. The current status of both the system located in Building 1120 and the system in Building 1124 will be documented in this section.

4.1 PDP-11/44 COMPUTER SYSTEM (BLDG. 1120)

4.1.1 PDP-11/44 Hardware

Table 4.1 shows the current status of the backplane of the PDP 11/44 minicomputer located in the RADC/EEV speech laboratory. The 11/44 processing unit replaces the previous 11/34 unit which has been relocated to another RADC/EEV laboratory. For each Unibus module present in the 11/44 system, Table 4.1 presents the board number(s), a description, memory address(es), interrupt vector location(s), and bus priority. Note that several of the modules listed are not presently interfaced into the system but are available for future use. On the other hand there are devices which are mentioned but are not actually available at the speech lab (e.g., a TM11 magtape controller); software device drivers have been generated for them nevertheless.

Figure 4.1 is a representation of the processors and peripheral devices currently at the speech lab. It illustrates the common communication path which they share—the Unibus. Figure 4.2 puts this hardware in a physical perspective as it illustrates the location of each device in the computer room.

The most significant change in hardware which took place during the contract period was the introduction of a PDP 11/44 processor. The advantages offered by the 11/44 in comparison with the previous 11/34 are:

1. increase in maximum physical address space from 18 to 22 bits.
2. 8 Kbytes of cache as compared with no cache memory on the 11/34.
3. intelligent ASCII console interface replaces the manual front panel controls on the 11/34.

TABLE 4.1
PDP 11/44 RSX-11M,V3.2 Hardware

The hardware consists of a PDP 11/44 with 1024K words (16 bits each) of MOS memory with Parity checking or an address space of 0-7640000 (octal) bytes, plus a UNIBUS addressing space of 8 Kbytes, i.e. 17760000-17777777(8).

BA11-AA unit, from right to left

Device	Controller	Function	Address	Vec	BR
		PDP 11/44 CPU			
-	KD11-Z/M7090	Console interface	777560	60	4
			777566	64	
		Line clock	777546	100	
	FP11-F/M7093	Floating		224	
	/M7094	Data Path			
	/M7095	Control			
	/M7097	Cache	777744-54		
	/M7098	Unibus interface			
	/M8743	1 Meg. byte of memory			
	/M8743	1 Meg. byte of memory			
	/M9202	Unibus connector			
TT1:	DL11-W/M7856	LA-36 via QUINTRELL	776520	330	4
		RS232-C, 300 baud	776524	334	4
TT2:	DL11-A/M7800YA	Tektronix 4015-1	775610	310	4
		@20ma,9.6Kbaud,self-clocked	775614	314	4
TT3:	DL11-W/M7856	ZENITH P.C.	775620	340	4
		EIA, 2400 BAUD			

UNIBUS to BA11-F =========>

BA11-F unit, from front to rear

Device	Controller	Function	Address	Vec	BR
SA:	DR11-C/M7860	16-BIT Parallel Interface	767760	410	5
		Digital data I/O SD350			
TT4:	DZ11-A/M7819	Speech Peripheral Bench	760010	350	5
		RS232-C, 4800 baud			
TT5:	"	VT-100 @ 9600 baud	"	"	"
TT6:	"	VT-100 @ 9600 baud	"	"	"
TT7:	"	M100, Cmptr Rm, @ 9600 baud	"	"	"
TT10:	"	Modem @ 300 baud	"	"	"
TT11:	"	M100, Sound Rm, @ 9600 baud	"	"	"
TT12:	"	VT-55 @ 9600 baud	"	"	"
TT13:	"	VT-100 @ 9600 baud	"	"	"
RL0:	RL11-AK/M7762	RL01-AK drive	774400	360	5
RL1:	"	RL01-AK drive	"	"	"
RL2:	"	RL02-AK drive	"	"	"
RL3:	"	RL02-AK drive	"	"	"
MP0:	HIC-11	MAP-300 Array Processor	766004	440	-
LP0:	LXY11/M7258	Printronix P300	777514	200	4
DR0:	Xylogics 650	Xylogics RM05 Emulation	776700	254	4

TABLE 4.1
PDP 11/44 RSX-11M,V3.2 Hardware

BA11-F unit, from front to rear (Continued)

Device	Controller	Function	Address	Vec	BR
SD0:	SD13209	Modified DR11-C's for	767700	300!	x
SD1:	"	Spectral Dynamics 360	767710	300!	x
SD2:		(looks like four devices to the			
SD3:		RSX-11M operating system)			
		UNIBUS to FPS AP-120B =========>			
	DDV11-C	LSI11 Backplane for:			
	DW11-B/M8217	UNIBUS/QBUS Converter for	-	-	-
IB0:	IBV11-A/M7954	Instrument Bus for SD-350	760150	420/430	x
		IEEE-488 Bus in/out =========>			
	DW11-B/M9401	QBUS Mirror Image	-	-	-
	TEV11/M9400YB	QBUS Terminator	-	-	-
	DW11-B/M9403	QBUS Connector	-	-	-
		FPS AP-120B			
AP0:	FPS #218	FPS Arith. Proc. AP-120B	776000	170	x

UNIBUS to BA11-E =========>

BA11-E unit, from front to rear

Device	Controller	Function	Address	Vec	BR
OMR	DC11-AB/M7821	Decision Inc. 6510	776500	300!	-
	" /M957,M594	Optical Mark Reader	776504	300!	-
	" /M105				
-	DB11-A/M7248	UNIBUS repeater	-	-	-
	" /M7213,M783				
	" /M784,M785,M930A				
		UNIBUS to CSP-30 S/N 29 =========>			
CSP30	1020A	CSP30#to#PDP#11/44 link	760030	370/374	-
		UNIBUS to CSP-30 S/N 20 =========>			
CSP30	1020A	CSP30#to#CSP30 link	760020	370/374	-
-	M930A	Passive UNIBUS terminator	-	-	-
		END OF UNIBUS			

*! Flags same register and/or vector address!!!

Special status devices

LOADABLE DRIVER, BUT CONTROLLER NOT INSTALLED:

Device	Controller	Function	Address	Vec	BR
MT0:	TM11	800 bpi Magtape	(772520)	224	5
CR0:	CR11-B/M8290?	Card Reader	(777160)	230	6
	CONTROLLER NOT CONNECTED:				
-	SD-13378	GPIB adapter for SD350-6, talks to IBV11-A	-	-	-
	LOADABLE DRIVER FOR PSEUDO-DEVICES:				
VD:	-	Virtual Disk driver	-	-	-

The following drivers are loadable rather than resident: AP:, CR:, SD:, VD:, MP:, MP:.

TABLE 4.1 (Continued)
PDP 11/44 RSX-11M,V3.2 Hardware

Controllers not Installed in System

Device	Controller	Function	Address	Vec	BR
-	DR11-C/M7860	16-bit Parallel Interface	767770	400	5
-	DR11-C/M7860	16-bit Parallel Interface	767760	410	5
-	M9301YF	Bootstrap (1000 bytes)	765000	-	-
	"	" (also 1000 bytes)	773000	-	-
-	RK11-D	Backplane and RK11 controller for RK05's (M7254, M7255, M7256, M7257)	777400	220	5
-	M930A	Passive UNIBUS terminator	-	-	-
-	M783	Bus transmitter	-	-	-
-	M785	Bus transceiver	-	-	-
-	M787	Bus Grant Continuity	-	-	-
-	M7820	Interrupt Control	-	-	-
-	M920	UNIBUS internal connector	-	-	-
-	W9042	FP11A Extender Board	-	-	-

TABLE 4.1 (Continued)
PDP 11/44 RSX-11M,V3.2 Hardware

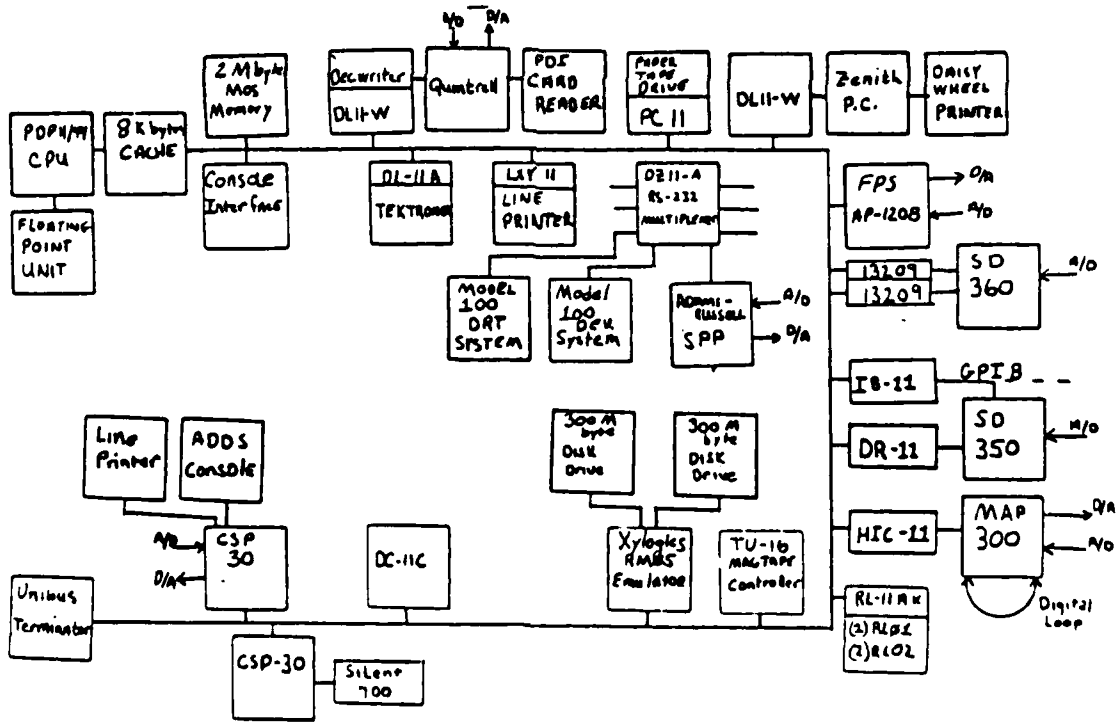


FIGURE 4.1
PDP 11/44 UNIBUS PERIPHERALS

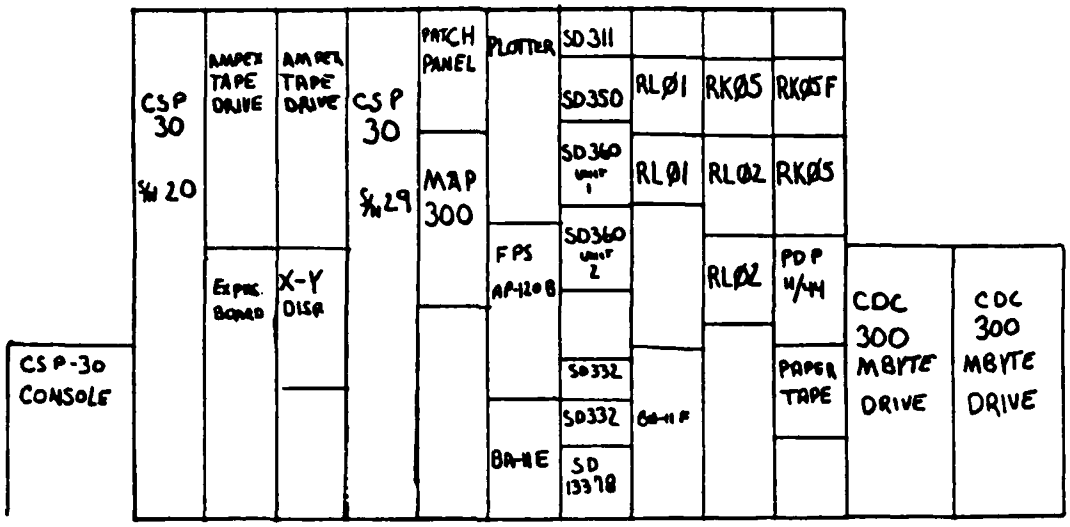


FIGURE 4.2
PHYSICAL LAYOUT OF COMPUTER FACILITY

The basic PDP 11 instruction set is maintained although the 11/44 does include a few more instructions useful for transitions between kernel, supervisor, and user modes. Most importantly, the previously used RSX-11M (V 3.2) operating system can be retained as well as all of the existing application software. The installation of the PDP 11/44 processor upgrade took place in December 1984. This process resulted in only a day and a half of system downtime. Physically, the box which houses the older PDP 11/34 was removed after the required peripheral interfaces were taken from the box's backplane. The new box which came with the processor upgrade had to be enhanced with a section of general purpose backplane in order to accept the peripheral interfaces removed from the old cpu box. The greatest installation delay was due to a faulty bus cable used to jumper from one section of backplane to the newly installed section. All boot roms from the old system were transferred to the new system. A new EIA interface was ordered for the LA36 (previously operating with a 20 ma protocol) to allow direct interface with the Console Driver of the PDP 11/44, which unlike the 11/34, has very limited front panel controls. These controls have been replaced by enhanced console commands entered from a peripheral (e.g., LA36 or VT100). The commands are described in Ref. 4.1.

Another hardware addition to the system is a second CDC-9766 300 Mbyte disk drive. This drive shares the Xylogics controller with the previous CDC-9766 drive. Problems were encountered during the installation process in accessing the new disk drive when its control bus (A cable) was daisy-chained off of the older drive. This problem was noted to be independent of the data bus port (on the controller) assigned to a particular drive. However, when the daisy-chain order was reversed such that the new drive's A cable was connected directly to the controller and the older drive daisy-chained off the new, both drives operated without error. The drives were left in this configuration with the new drive assigned to port 0 and the old drive assigned to port 1.

Several of the serial interface links to the system have been redirected. During the 11/44 upgrade, a DL-11W interface module became available since the new processor has an internal console communications package. These new links are tied to such things as a Zenith PC, a Model-100 development system, programmable filters and speech processing peripherals. The details of these interfacings are presented in Chapter 6 of this report describing software tool developments. Also mentioned in that section is a custom logic circuit added to the system for the purpose of interfacing the PDP-11/44 with the SD-350 Signal Analyzer.

4.1.2 PDP 11/44 System Software

A new RSX-11M operating system (OS) was generated for the recently installed PDP 11/44 which takes advantage of the new resources provided by this machine in contrast to the replaced PDP 11/34. In particular, the new OS recognizes a main memory address space of 2 megabytes, utilizes 8 Kbytes of high speed cache memory, and recognizes a second 300 megabyte disk drive.

All source files and command files used to build the new OS are contained on a virtual disk with the volume name of "Sysgen". The SYSSAVE.CMD file used to control the Sysgen process is located on UIC [200,200]. A note of caution regarding this sysgen is in order. The RL02 controller on this system has a non-standard interrupt vector address. This address is 360 while the DEC-standard and sysgen default address is 160. Another special feature to be noted in regards to this command file concerns the assignment of "9766" to the symbols $DR0 and $DR1 which indicate that the drives to be recognized as a "DR:" device are actually CDC 9766's. This value is legal because the sysgen command file responsible for setting up the data structures and command files used in building the RSX peripherals, SGNPER.CMD, includes a patch provided by Xylogics Inc. which builds an appropriate driver data structure (Unit Control Block) for this size drive. Patches were also required for the SAVSUB module included in the [1,24] SAV.OLB file which sizes the disk properly, and for the INIBAD module located in [1,24] INI.OLB which allows the reading of the manufactures bad block file on disks during disk initializations.

A couple of edits were required for some of the command files used to build system tasks. The command file used to build the indirect command processing task INIBLD.CMD was edited such that parent-offspring tasking is permitted within this task as required to enable the suppression of command line displays while executing an indirect command file (i.e., the ENABLE QUIET option). BIGIND.CMD was edited and rebuilt in the same manner. The BYE system task was also built using non-standard source files. These files include additional "user subroutines" which are executed when a user logs off:

1. Spawning of a "DVD /ALL/NM" command to insure all of the user's virtual disks are deallocated from the virtual disk driver.
2. Checks for the /NM switch included with the BYE command which suppresses the display of logout messages and clears the screen of the logout terminal.

The system task builder (TKB) was also rebuilt with global patch statements included in the build command file to select TKB switch defaults to match the characteristics of the current system (e.g., tasks use floating point coprocessor unless otherwise indicated).

The user-written, loadable device drivers on the system were all rebuilt to map the new OS. These drivers include:

1. AP: for the FPS 120b array processor
2. MP: for the MAP 300 array processor
3. SD: for the Signal Dynamics Spectrum Analyzer
4. VD: for the Virtual Disk system

Following a successful pass through the RSX Sysgen process, the resulting RSX-11M system image was configured for booting via the Virtual Monitor Console Routine (VMR). VMR provides the capability to execute MCR commands that are directed to the disk resident image of the system. The indirect command file used during the VMR process is called [1,54]NEWVMR.CMD and can be found on the Sysgen virtual disk as well as on the finished system disk. A major difference between this VMR process and the one used on the previous OS used on the PDP 11/34 concerns the establishment of device commons as subpartitions of the main partition IOPAG. It was found that only one subpartition could be established at a time within IOPAG as attempts to 'set' a second partition were met with a VMR alignment error. A solution to this problem is simply to remove the main partition IOPAG and establish each separate device common as an individual main partition of type 'dev'. The other changes involve the SET commands required to configure the two terminal lines, TT10 and TT7. These channels were apparently not in use at the time of the previous system generations and their initial states were always defined during system startup via the STARTUP.CMD file. Their initial states are fixed in the system image and no longer need to be set at startup. The STARTUP.CMD file on [1,2] was also edited to reflect these changes.

There have been revisions to the Log-in Accounts on the system. Table 4.2 lists the currently active system, backup, and research and development accounts on the system. The additional accounts allow access to newly created virtual disks (e.g., the ANDVT disk), or provide access to command files to provide operations for which a user may not know all of the required sequences of commands (e.g., the RADC/DRTLD provides a means of downloading code and data to a Model 100).

ACCOUNT/PASSWORD	LOGIN UIC	PURPOSE
	—————— SYSTEM ACCOUNTS ——————	
RADC/ERRLOG	[1,6]	Creates error listing
RADC/SHUTUP	[1,7]	Shuts down system
RADC/NOCSP	[3,2]	Turns off CSP-30 prog.
RADC/CSP	[3,3]	Starts CSP-30 prog.
RADC/MAPUP	[3,31]	Inits. MAP-300 exec.
RADC/DRTLD	[3,16]	Model 100 downloading
RADC/CLRQUE	[3,17]	Clear print queue
	—————— BACKUP ACCOUNTS ——————	
NAME/BACKUP	[3,**]	Backs up NAME.DSK
	—————— R&D ACCOUNTS ——————	
Q/xxxx	[7,202]	Downloads code to Quintrells
Q/xxxx	[7,203]	Accesses QUINTREL.DSK
MAP/xxxx	[7,202]	Downloads code to MAP-300
AGP/xxxx	[7,300]	Runs SD360 software
ANS/xxxx	[7,350]	Accesses Acoustic Noise Disk
DRT/xxxx	[10,200]	Accesses DRT Data Base
SSP/xxxx	[7,202]	Access to Adams-Russell SSP's
ANDVT/xxxx	[7,330]	Accesses ANDVT.DSK

TABLE 4.2
LOGIN ACCOUNTS FOR PDP 11/44 SYSTEM

4.2 PDP-11/34 COMPUTER SYSTEM (BLDG. 1124)

4.2.1 PDP-11/34 Hardware

Following the installation of the new PDP 11/44 at the Speech Lab, the replaced 11/34 was transported to the speech lab in Building 1124 where it replaced the disabled PDP 11/20. Also installed were two new RL02 disk drives permitting the system to run with a full version of RSX11M. A problem with one of the 9 track mag tapes on that system was also alleviated during the installation by a new fuse and the securing of a power line on the drive. The peripherals used with the 11/20 were retained. A list of modules connected to this system's Unibus are presented in Table 4.3.

TABLE 4.3
PDP 11/34 Hardware

The hardware consists of a PDP 11/34 with 124K words (16 bits each) of MOS memory with Parity checking or an address space of 0-757777(8) bytes, plus a UNIBUS addressing space of 20(8) Kbytes, i.e. 760000-777777(8).

Device	Controller	Function	Address	Vector	BR
-	KD11-EA M8266	CPU, board 2	-	-	-
-	"	CPU, board 1	-	-	-
		with Memory	772300...356	250	-
		Management at	777572...656		
-	FP11 M8267	Floating Point	-	244	-
-	MR11-EA M9312	Boot diags,Console Emulator	765000	-	-
	"	" DLn:	773000	-	-
		and			
-	KY11-LB M7859	Programmers Console with			
		Console Switch Register	777570	-	-
-	MS11-LD M7891	MOS memory with	0-757777	-	-
-	" (M7850)	Parity Controller	772100	114	-
TT0:	DL11-W M7856	VT-52 Console	777560	60	4
		RS232-C			
TT1:	DL11-W M7856	LA-36	777530	300	4
		@20 ma, 300 baud	776524	334	4
PR0:	PC11 M7810	Paper Tape Reader	777550	70	4
PP0:	"	Paper Tape Punch	777554	74	4
RL0:	RL11-AK M7762	RL02-AK drive	774400	160	5
RL1:	"	RL02-AK drive	"	"	"
MT0:	TM11	800 bpi Magtape	772520	224	5
************** Devices available without Drivers****************					
	KW11-K	PROG. CLOCK	170404	444	-
	AD11-K	A/D CONVERTER	170400	340	-
	AA11-K	D/A CONVERTER	170416	360	-

TABLE 4.3
PDP 11/34 Hardware

4.2.2 PDP 11/34 System Software

A Sysgen for the PDP 11/34 was carried out in order to provide a RSX-11M operating system configured for the peripheral devices available and the user load expected. The major difference between this new operating system and the one previously used on the PDP 11/34 as it ran at the Speech Lab is in regards to the peripheral devices known to the OS, their address locations, and the types of loadable device drivers available to control these devices. A system disk was prepared (RL02) containing a bootable system image and the required system tasks needed to get RSX up and running. The devices supported by the new OS are:

1. RL02 disk drives
2. 9 track Mag tapes
3. two terminals
4. paper punch and reader

The A/D, D/A and the programmable clock subsystems are also available through device commons. The command files and resulting task files for this sysgen can be found on the virtual disk GEN1124, which in turn, is located on the 300 Mbyte disk pack labeled as being the system disk for the old 11/34 system used in the Speech Lab.

4.2.3 PDP-11/34 Application Software

Currently the dedicated application for the PDP 11/34 is the collection of EPL and Speech/Noise ratio data from DRT source tapes (Ref. 4.2). This application software was revised such that the program no longer needs to be run in the absence of an operating system (i.e., stand alone), but is able to execute as a RSX-11M task. This design makes available all of the resources of the operating system and avoids the inconvenience of having to boot a stand alone task and then reboot the RSX11M system every time the program is needed. The major changes required for this redesign were:

1. Use of QIO system calls for terminal I/O
2. Establishment of a device commons for accessing the command/status registers of the A/D and Clock peripherals.

Other changes involve the disabling of the system clock on the PDP 11/34 during the 40 second period of data acquisition. This unorthodox procedure allows the task to have complete and sole use of the CPU as needed for executing the closely timed real time data acquisition loop. The disabling of the system clock will cause any other task currently executing to enter a 'wait state' for 40 seconds before being rescheduled for CPU service. The use of system directives $Gtim and $Spwn allow continual correction of the system clock which is disrupted during real-time data acquisition. The new software version has been tested at the speech lab facilities and has been used in validating previous EPL measurements made using the PDP 11/20 system in Building 1124.

The task image for this task is located in a file [200,200]RSXV.TSK on the RL02 system disk. A command file has been created to do the installation of this task along with the device common, A2DCOM. The task can be initiated by a user with UIC set to [200,200] by typing "@RUN". Once the program is running, the user instructions are the same as with the earlier stand-alone version except for the fact that the user can type Ctrl-Z to exit out of the program and return to the operating system.

CHAPTER 5

ALGORITHM RESEARCH AND IMPLEMENTATION

Improvements to the LPC-10 speech compression algorithm have been implemented at the RADC/EEV Speech Lab during the contract period. These implementations provide tools for comparative algorithm research and also provide RADC/EEV personnel with detailed information regarding the advancements made toward the development of a new standard narrowband speech compression algorithm for use by the DOD. Other efforts in this area consist of making available to the Speech Lab researchers MAP-300 versions of compression algorithms for which the software was provided by sources outside of RADC/EEV.

5.1 LPC-10 IMPROVEMENTS

The LPC-10 improvements implemented by ARCON form a subset of the improvements being offered by G. Kang and S. Everett of the Naval Research Laboratory (Refs. 5.1 and 5.2). All of the implementations to date have been non-real-time on a PDP-11/44 with some of the applications making use of an attached math processor—a MAP 300. These improvements were integrated into the available Interactive Laboratory System (ILS) software package which includes LPC analysis and synthesis modules and which uses the same data I/O formats as does the speech data base software on the RADC/EEV system.

5.1.1 Onset Detection And Window Alignment

LPC voice processors are known to distort voiced onsets such as /b/, /d/, and /g/ resulting in degradation of consonant intelligibility. This distortion is believed to be related to the misplacement of analysis windows relative to actual onsets. A misplaced window can result in poor abstraction of speech parameters for a particular frame of data because the data actually contains information regarding two discrete phonetic events. Slurred and fuzzy speech is said to be the result of this faulty analysis. The onset detector suggested by Kang and Everett is based on the monitoring of single-step forward and backward predictors (using a minimum least squared error criterion) for the preemphasized speech waveform. Non-stationary segments of the speech waveform (i.e., onsets) will be associated with larger differences between the forward and backward predictors compared to their differences noted during segments of stationary speech. Also, a significant change in either the forward or backward predictor alone, in comparison with its recent history (16 samples), is taken as an indication of a change in the statistics of the speech waveform and thus is regarded as a phonetic transition.

Figure 5.1 illustrates the performance of this onset detection algorithm as included in the ILS module available for displaying sampled waveforms on a Tektronix terminal.

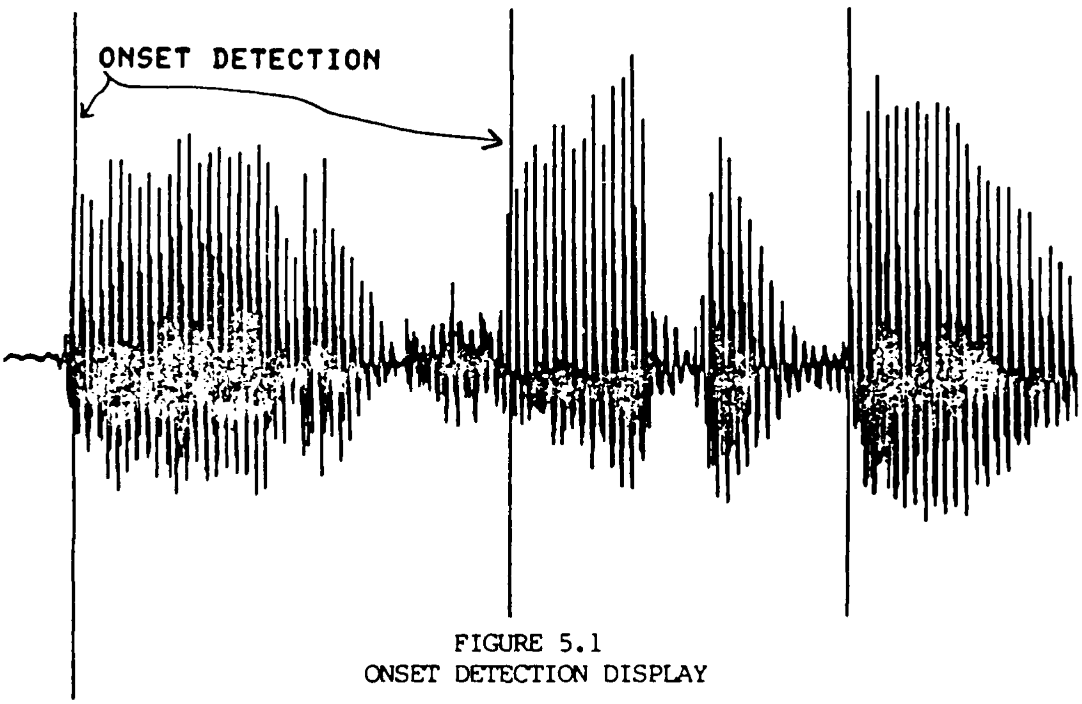


FIGURE 5.1
ONSET DETECTION DISPLAY

The locations of onsets are marked by a vertical line running the full length of the display terminal's screen. The algorithm is not demanding in terms of computation time or memory requirements. A total of 7 multiplies, 2 divides, and 4 addition/subtraction operations are required per sample. Eleven state variables are required for the onset detector. The source code for the ONSET subroutine can be found on the REPT86 disk.

5.1.2 Modified LPC Analysis For Sustained Vowels

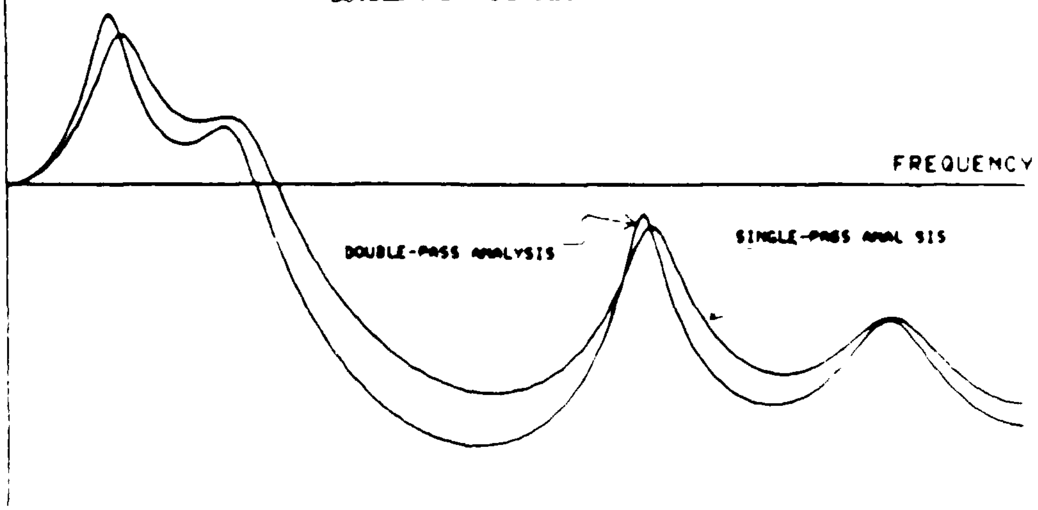
In order to enhance the transmission of sustained vowel sounds using LPC, Kang and Everett have suggested a two-pass analysis. The goal of the second pass is to provide a more accurate modeling of the actual speech spectrum by the all-pole filter used in LPC to represent the vocal tract. This filter's coefficients are determined by solving a set of linear prediction equations for which a given speech sample is predicted by a weighted sum of past speech samples. Kang and Everett note that the prediction of the current speech sample by a weighted sum of past speech samples is valid except when the waveform is disturbed by glottis excitation at the beginning of each pitch period.

The inclusion of the equations associated with these disruptions "leads to broadened resonant bandwidths that make the synthesized speech fuzzy." The suggested improvement involves the deletion of equations for which the residual obtained from the first pass analysis is greater than twice the RMS of the total residual for the analysis frame. A second array of predictor coefficients is obtained from the solution of the matrix equation defined by the reduced set of linear equations.

Software has been written to study the effects of this suggested improvement. This task, KEA, obtains speech data from files in the format used by the ILS routines and outputs ILS "analysis" files. The ILS task "API" which performs a standard LPC analysis and pitch detection formed the foundation of this new task. The input commands for KEA are the same as for API. It was decided to use the Macro Array Processor (MAP) for this task due the heavy computational load required for a dual LPC analysis. The MAP was used to perform the "loading" of the matrices involved in the system of linear equations and the solving of these equations required to obtain the prediction coefficients. The "covariance" LPC analysis method was used. The cepstral pitch detection method used by the original API task was retained.

The ILS task FPL allows a comparison of the vocal tract spectrum supplied by the all-pole LPC filter. The sharpening of the formants for sustained vowels due to the enhanced analysis efforts were noted as shown in Figure 5.2.

FIGURE 5.2
COMPARISON OF SPECTRAL ENVELOPES FOR
DOUBLE-PASS AND STANDARD LPC-ANALYSIS



Statistics regarding the bandwidths of the first four formants for 100 frames of speech were obtained from both a standard "single pass" and an improved "double pass" analysis. The resulting data shown in Table 5.1 also supports the contention that the two pass analysis does produce all-pole models of the vocal tract with narrower formants.

	ENHANCED ANALYSIS (DOUBLE-PASS ANALYSIS)		UNENHANCED ANALYSIS	
FORMANT	MEAN	STAND. DEV.	MEAN	STAND. DEV.
1	2.9632E+02	2.4562E+02	3.4419E+02	2.5579E+02
2	3.6167E+02	2.4418E+02	3.7282E+02	2.3128E+02
3	3.4433E+02	2.2957E+02	4.2530E+02	2.3968E+02
4	3.1627E+02	2.3909E+02	3.5715E+02	2.2374E+02

TABLE 5.1
FORMANT BANDWIDTH STATISTICS

NSA researchers have chosen not to include the two-pass analysis idea in the LPC-10E algorithm (Ref 5.3). Their rationale is based upon the observation that the variability of a formant's location on the frequency dimension is increased when equations are deleted from the standard set used in solving for prediction coefficients. Data collected at RADC corroborates this statement. Statistics collected on the first four formants for the same 100 frames of speech mentioned above indicate that formant frequencies obtained from a two-pass analysis demonstrate more variability compared to those obtained from a standard analysis (see Table 5.2). An informal survey of listeners at RADC who were presented with DAM sentences processed with both the standard and the two-pass analysis methods did not indicate that a perceptual improvement exists.

	ENHANCED ANALYSIS (DOUBLE-PASS ANALYSIS)		UNENHANCED ANALYSIS	
FORMANT	MEAN	STAND. DEV.	MEAN	STAND. DEV.
1	7.0324E+02	4.4113E+02	7.7012E+02	4.0022E+02
2	1.5045E+03	6.4725E+02	1.5909E+03	5.0548E+02
3	2.3420E+03	8.5419E+02	2.5978E+03	4.8777E+02
4	2.9733E+03	1.3075E+03	3.1166E+03	1.1235E+03

TABLE 5.2
FORMANT FREQUENCY STATISTICS

[illegible] is available on the REPT86 disk for the KEA task. A few [illegible] order regarding the use of SNAP II functions for the MAP [illegible]. The MAP executive must include the Extended Array [illegible] which include the functions used to manipulate matrices. [illegible] are not included in the executive loaded with the DR:[1,4]MAPUP.CMD file, but can be found in a binary file called DR:[7,160]EAF33.BIN. A command file, DR:[7,160]MAPUP.CMD, exists which loads the appropriate executive. A second note is that problems were encountered when attempting to use memory-resident overlays with code containing SNAP II function calls. In particular, functions which performed HOST-MAP I/O did not work as expected. The problem is due to the fact that this I/O is done using Direct Memory Access (DMA), and in the case of the MAP transmitting data to Host memory, no consideration is given to any Host memory remapping which may have occurred during the overlay process. A solution to the problem is to allocate all program variables which are to receive data from the MAP in the root segment of the task which is never remapped. A third note is in regards to the incorrect documentation of the SNAP II function called MFS which performs a triangular factorization of a symmetric matrix as required for a Cholesky solution of the matrix equations in the LPC analysis process. This function is documented as requiring only two arguments while it was necessary to add two extra "dummy" arguments to make it work.

5.1.3 Spectral Shaping Of The Excitation Signal

These improvements focus upon the generation of a more accurate excitation signal for the LPC synthesis filter. The term "more accurate" here refers to the degree of agreement between the excitation signal to be used to drive the synthesis filter and the ideal excitation signal represented by the residual from the LPC analysis filter. Past LPC synthesis algorithms have used impulse sequences (with periodicity determined by a pitch estimate) for voiced excitation and "random" sequences for unvoiced excitation. However, it is realized that these simple sequences only grossly resemble the actual residual signal. Amplitude and phase spectral shaping are used to generate more accurate excitation signals. Figure 5.3 compares the unimproved and improved voiced excitation signals.

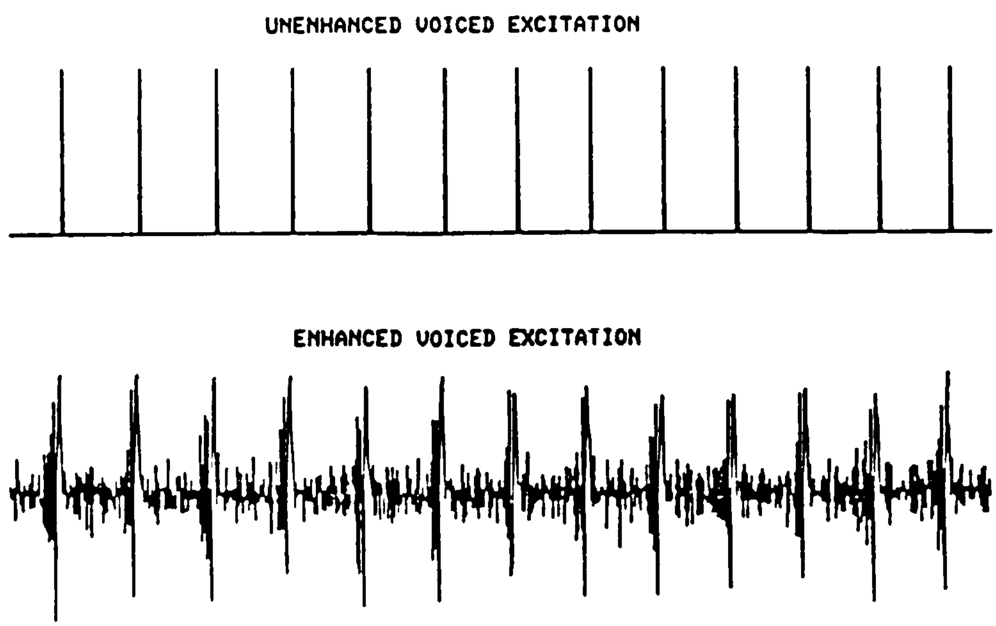


FIGURE 5.3
COMPARISON OF UNIMPROVED AND
IMPROVED VOICED EXCITATION SIGNALS

Amplitude Shaping - The spectral amplitude of the residual obtained from a frame of voiced speech is not completely flat as is the spectrum for the simplistic impulse response traditionally used to drive the synthesis filter during voiced frames. Kang and Everett have indicated a need for an excitation signal which contains spectral amplitude shaping which will vary from voiced frame to voiced frame. They have suggested that the spectral amplitude of the excitation signal be determined by the shape of the spectral amplitude of the input speech which is modeled by the all-pole LPC filter during the LPC analysis. They report that without some spectral amplitude shaping, "the synthesized speech tends to sound fuzzy and lacking in clarity." The transmitted reflection coefficients used for vocal tract modeling can be used to provide the required spectral amplitude shaping. The shaping can be implemented by way of filtering (all-pole) a signal with an initial flat magnitude spectrum. The filter coefficients are proportional to the coefficients used to model the vocal tract with the constant of proportionality being a function of the ratio of the residual RMS to the speech RMS. Thus the residual formant peaks will become smaller as the residual RMS decreases for portions of the speech waveform in which the inverse filter becomes more efficient (i.e., front vowels, murmurs and nasals).

Phase Shaping - The spectral phase components of the excitation signal are also selected so as to generate an excitation signal more similar to the residual signal. According to Kang and Everett, the phase spectrum for voiced excitation is made up of three parts. The first part is stationary and is a quadratic function of frequency as determined by previous research regarding the spectral phase of the residual signal. The remaining two parts both contain random variables. One part is related to pitch-epoch variation, or jitter, caused by irregularities in vocal cord movement. The other random part is related to period-to-period waveform variation caused by turbulent air flow from the lungs.

Unvoiced Plosive Excitation - In order to more accurately represent the sudden burst of energy associated with unvoiced plosives (e.g., /p/,/t/,/k/), Kang and Everett have suggested adding random spikes to the excitation signal for unvoiced frames containing plosives. The conventional random-number generated unvoiced excitation signal works fine for fricative sounds. However, Kang and Everett report "this excitation is not satisfactory for generating burst sounds. The onsets of these sounds generate large spikes in the prediction residuals, but the excitation signal conventionally used to synthesize them is still stationary noise. As a result CAT is often heard as HAT, and TICK may sound like THICK or SICK."

The ILS program called SNS which performs a pitch synchronous synthesis of speech based upon parameters obtained from LPC analysis, was revised to include these improvements. The revised program is referred to as KES ("Kang and Everett Synthesis") and its task image exists on the virtual disk KANG at UIC [7,161]. The format of the input parameters for KES are the same as for SNS. The secondary file must be an analysis file and the primary file will become the sampled data file containing the synthesized speech waveform following the execution of KES. The ILS subroutine SYNPF has been replaced by the subroutine SYN located in the file SYN.FTN. A new subroutine called VOICEX has been written to perform the generation of the improved voiced excitation signal (this subroutine is in the file KESYN.FTN). All source code for KES can be found on the REPT86 disk.

The excitation signal used in KES is generated using an inverse discrete Fourier transform. The spectral amplitude and phase components used for the transform are determined on a pitch period basis for voiced frames and are determined once per frame for unvoiced frames. The spectral amplitude of the excitation signal is determined by the shape of the spectral amplitude of the input speech which is modeled by the all-pole LPC filter during the LPC analysis. The transmitted reflection coefficients used for vocal tract modeling provide the required spectral amplitude shaping. The shaping is implemented in KES by way of filtering (all-pole) a signal with an initial flat magnitude spectrum. The filter coefficients are proportional to the coefficients used to model the vocal tract with the constant of proportionality being a function of the ratio of the residual RMS to the speech RMS. This filtering is included for both the voiced and the unvoiced excitation signals. All of the spectral phase components for the voiced excitation signal as discussed above are included in the subroutine VOICEX which returns a generated excitation signal sample resulting from the inverse Fourier transform of a spectrum with these phase components.

The detection of unvoiced plosives in KES is made by monitoring the change in speech RMS from one unvoiced frame to the next. A ratio is formed of the speech RMS for the current frame and the speech RMS for the previous frame. A ratio of 4 or more indicates the presence of an unvoiced plosive. The amplitude of the spikes added to the excitation signal for such a frame is made proportional to this ratio of speech RMS. The spikes are randomly added to the excitation signal with the probability of a particular excitation signal sample having a spike being 0.05. The unvoiced excitation signal is generated in the subroutine SYN.

5.1.4 Input/Output Bandwidth Expansion

It is recognized that stop consonants and voiceless fricatives contain frequency components extending past the 4 kHz bandlimit traditionally set for LPC processors. As a result, intelligibility and speech quality are degraded. Kang and Everett have suggested that the normally avoided aliasing affect be exploited in order to spread the sibilant sound spectra past the 4 kHz boundary. A more accurate spectral representation of fricatives can be obtained by folding the 2 to 4 kHz spectral contents up to the 4 to 6 kHz range. Since there is little distinctive formant information in the sibilant sounds, the spectrum between 2 and 4 kHz is similar to that between 4 and 6 kHz. This spectral folding is implemented by simple interpolation of the output waveform. The output sampling frequency is doubled and zeros are added for every other sample. A reconstruction low-pass filter with a gentle roll-off is desired in order to retain the higher frequency components (i.e., those between 4 and 6 kHz). This filtering subsystem may be a combination of digital and analog filters.

The output spectral folding technique has been implemented by ARCON using the D/A capabilities of the MAP-300. A revised MAPOUT program, KESOUT, has been developed (task image resides on Kang [7,161]) which performs this interpolation of the output waveform. The source code for KESOUT can be found on the REPT86 disk. The synthesized speech data file is output on the MAP's AOM at twice the original sampling frequency with every other sample being a zero. All of the interpolation process takes place in the MAP; the input file does not have to be preprocessed. A linear phase, finite impulse response filter was designed using the tools available in ILS. This filter provides a gradual roll-off of the spectral energy above 4 kHz and gently shapes the folded formants in the 4-6 kHz spectral region. An analog low-pass filter is recommended with cutoff frequency set around 7 kHz for final reconstruction of the output waveform. Figure 5.4 provides a spectral display of a synthesized speech signal generated with KESOUT.

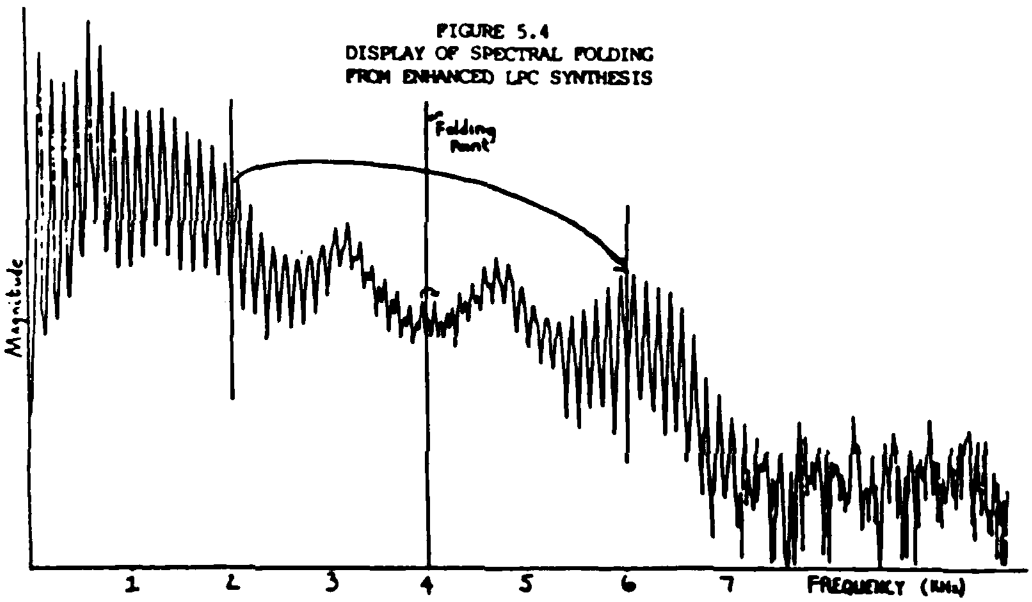


FIGURE 5.4
DISPLAY OF SPECTRAL FOLDING
FROM ENHANCED LPC SYNTHESIS

The MAP must be loaded with the DR:[1,54]MAPUP command file before executing KESOUT. The extended array function library must also be loaded into the MAP before execution. This is done by running the MAPLOAD utility and indicating the load file to be DR:[7,160]EAF33.[illegible] when prompted for a binary file. *QIOLIB* routines are used to access the speech data file just as with the original MAPOUT utility.

5.2 MAP-300 IMPLEMENTATIONS

Three speech compression algorithms are currently available as real-time, full-duplex MAP-300 implementations. The programming of these implementations was done outside of the Speech Lab. The MAP-300's analog input and output subsystems are used to input and output speech waveform signals with the output signal representing a processed version of the input waveform. All algorithms take advantage of the 32-bit floating point numerical representation offered by the MAP-300. A Continuously Variable Slope Delta Modulation (CVSD) algorithm is available with user selectable parameters: Frequency, Stepsize, Predictor Time Constant, and Min and Max CVSD Stepsize. A 9.6 kilobits/second Adaptive Predictive Coder with Segmented Quantization (APC/SQ) and a 2.4 kilobits/second, tenth-order Linear Predictive Coder (LPC-10, Version 44) are also available (Ref. 5.4). Optional features for these latter two implementations include software generated channel error simulation and a speech coder status display which is useful for determining such things as the peak input speech level.

The CVSD and APC/SQ implementations have performed successfully on the RADC/EEV Speech Laboratory's MAP-300 system. However, the LPC-10 algorithm does not work properly on this system as synthesized output speech frames are "lost" resulting in gaps in the output speech waveform. It is known that the MAP-300's used on the DCEC system (for which these algorithms were developed) have a faster third bus memory compared to the RADC/EEV machine. Furthermore, during intensive arithmetic operations on a MAP-300 in which both of the available arithmetic units are used, the execution time becomes limited by the memory bandwidth. This hypothesized explanation for the implementation's failure can be tested once the MAP-300 units received from DCEC are operational at the RADC/EEV Speech Processing Facility.

A user can execute any of these algorithms on the MAP by logging into an account set up specifically for this purpose. The account is accessed as MAP/LOAD with all of the required executable code located on DR:[7,204]. A login command file for this account will prompt the user for the type of algorithm to be loaded into the MAP for execution. The specific user instructions for the LPC10 and APC/SQ algorithms can be found in Ref. 5.4.

CHAPTER 6

SOFTWARE TOOLS

New software tools have been added to the RADC/EEV Speech Processing Computer Facility. These tools extend the system's research capabilities for the analysis, synthesis, storage, and display of speech related data. The transfer of control and data between the PDP-11 host computer and the Adams-Russell speech processing peripheral, and the programmable filters is now possible. To a limited extent, the resources offered by the SD-350 spectrum analyzer have also been made remotely controllable by the host computer. An extensive set of programs for the analysis and display of speech signals has been added in the form of the Interactive Laboratory System (ILS) package. The pre-existing speech data base has been made compatible with the ILS software.

6.1 SD-350 INTERFACING

The SD-350 is a digital spectrum analyzer providing real-time spectral displays of analog input signals (Ref 6.1). A user has control, via front panel switches, over such parameters as FFT length, bandwidth, and averaging modes. It was determined that better use of this resource could be made if a user were able to obtain the numerical spectrum information from this device for storage and later analysis on the host computer system. Furthermore, it would be desirable to have remote control of this spectrum data acquisition process. Thus a project was defined to provide control and communication capabilities for the SD-350 with the following specific objectives:

1. remote monitoring and control of SD-350 front panel parameters
2. digital spectrum data I/O between the SD-350 and PDP-11.
3. digital time data output from the PDP-11 to the SD-350.

The first two objectives could be achieved by using the General Purpose Instrument Bus (GPIB) control unit provided by Spectral Dynamics. The GPIB does not provide the capability to transmit digital time data to the analyzer. However, there is a data input port on the back of the SD-350 for this purpose. Digital interface circuitry was developed in order to provide a communication channel between the SD-350 and the PDP-11 by which time data could be sent to the analyzer's input memory for processing. A report on this interfacing technique will first be presented and then the work which has been done involving the GPIB will be discussed.

6.1.1 Digital Time Data Transfers

The SD-350 is capable of receiving 10 bit digital data in a 2's complement format. Several control lines are required in addition to the 10 data lines in order for external data to be loaded into the SD-350's input memory. Digital interface circuitry was designed and built by ARCON personnel to provide the following capabilities:

1. Tri-stated data lines which go to the high impedance state when external data input is disabled (i.e., the SD-350 takes time data for processing from its internal A/D)
2. Enable external data and enable external sample signals
3. External input memory load signal (strobes input data) and external sample clock signals
4. External memory hold signal which allows the contents of the SD-350's input memory to be held constant

This circuitry was implemented with TTL logic on a small vector board which was located in the backplane of the BF11-FD box of the PDP-11 computer system (see the schematic in Figure 6.1.). Power for this circuitry comes from the SD-350. The output signals from this circuitry are connected to the back panel of the SD-350 with a 16 line ribbon cable and one separate line which was added during the later development stages (EXT MEM HOLD).

Data and control signals from the PDP-11 are provided by a DR-11C parallel interface module (CSR=17776510) which is connected to the interface circuitry by a short ribbon cable. The two control bits provided by the DR-11C are used to select the control mode of the SD-350. These bits are referred to as CSR0 and CSR1 (bits 0 and 1, respectively, of the DR-11C's CSR). The possible settings of these control bits are:

CSR0	CSR1	VALUE WRITTEN TO DR11 CSR	MODE
0	0	0	EXTERNAL DATA ENTRY DISABLED
0	1	1	EXTERNAL DATA ENTRY ENABLED
1	0	2	EXT DAT ENABLED WITH INPUT MEMORY HELD
1	1	3	EXTERNAL DATA ENTRY DISABLED

The logic used to control these lines was designed such that the normal (non-remote) operation of the SD-350 would not be disabled when

1. the computer system is not powered up,
2. the DR-11C is initialized upon power-up.

Thus explicit commands must be given before the SD-350 is forced into an external data entry mode. Data to be sent to the SD-350 is written into the DR11's data buffer causing the data strobe signal to be generated as needed by the SD-350 for reading the 10 data lines. The rate at which the data is sent to the SD-350 is crucial as this is the "sampling rate" from which the spectral frequency units provided by the SD-350 are derived. A spectral shift along the frequency dimension is expected if this sampling rate does not match the actual sampling frequency used to originally obtain the digital time data. There is no programmable clock on the PDP-11 system at the Lab which

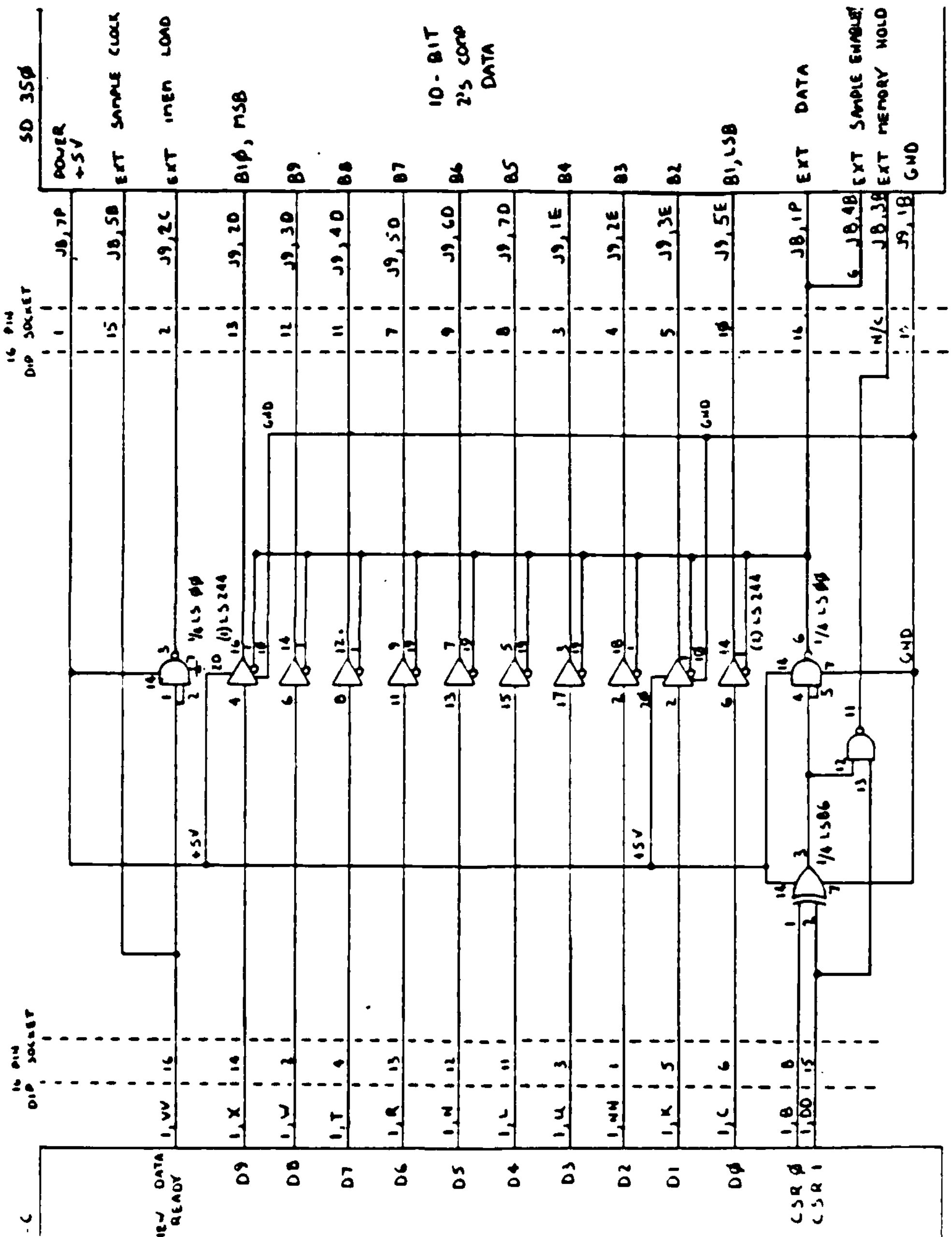


FIGURE 6.1
SCHEMATICS FOR SD-350 DIGITAL TIME DATA INTERFACE

could be used to provide a timing interrupt for determining the time at which to send another data word to the SD-350. Instead, a wait loop which provides a constant sampling frequency of 8 kHz was included in the Fortran code written to test this new interface. A check of the generated timing pulses using an oscilloscope indicated that this sampling interval was within 5 micro seconds of the original.

A program to exercise this digital interface was written under the name of SPS ("speech spectrum") and for which a task image can be found on the IFACE virtual disk at UIC [200,200]. Source code for SPS can be found on the REPT86 disk. This program prompts the user for an unprocessed speech data file and for a starting block number in the file. The 2048 contiguous data points from the starting block onward are sent to the SD-350 for analysis. The data is scaled as needed to fit within the 10-bit constraints of the SD-350. The user has the option of holding the spectral display or not. Since no control of the SD-350's front panel is assumed, the user is responsible for manually setting the transform size parameter to 2048. Another program available for sending digital data to the SD-350 is DSB located on the IFACE virtual disk. This program is described further below

6.1.2 GPIB Interfacing

As previously mentioned, a potential pathway for the transfer of control and data between the PDP-11 and the SD-350 is an IEEE-488 or GPIB bus. Spectral Dynamics provides a GPIB Adapter (Model 13378) which ties the SD-350 into the IEEE-488 bus (Ref. 6.2). The Adapter consists of IEEE-488 interface circuitry, circuitry to drive the external control and data lines coming out of the SD-350, a microprocessor, and RAM storage for buffering the data during transfer processes. The Adapter provides the following interfacing capabilities to the IEEE Bus:

1. Read switch position of designated SD-350 front panel controls.
2. Read real-time or averaged data from the SD-350.
3. Read data from the SD-350's AUX MEMORY.
4. Control switch position of designated SD-350 front panel controls.
5. Supply data to SD-350 AUX MEMORY.

The PDP-11 is connected to the IEEE-488 bus by way of an IBV11-A "LSI-11/Instrument Bus Interface" module (Ref. 6.3). This module is mapped into the PDP-11 I/O page address space and serves as a controller for the IEEE-488 bus. As noted in the name of this device, it is intended to be used with a Q-bus as opposed to a Unibus as found on the PDP-11. The solution to this problem is the introduction of a Unibus to Q-bus converter as found on the Speech Lab's system. This conversion is transparent to the systems programmer.

Software source material provided by NASA for building an IBV11-A bus controller "device driver" resides on the REPT86 disk in files IEDRV.MAC and IETAB.MAC. A driver was built from this source code without modification and "loaded" into the system with the device mnemonic "IE:". The IEDRV, and IETAB object modules were included in the [1,24]RSX11M.OLB library. I/O requests to the bus controller can now be made with QIO system directives. Experimental I/O requests were made to send commands to the GPIB module. The GPIB controller did respond to a Master Clear request (function code [illegible]) as indicated by the "RESET" light-emitting diode on the GPIB's front panel. The controller did not respond to requests to "address as listener" or "address as talker". The problem occurs independently of the IEEE-device address set for the Adapter by dip switches on the back of the unit and independently of the presence or absence of cables interfacing the Adapter to the SD-350.

A more detailed analysis of the I/O problem was made by using the [illegible] utility which allows a privileged user to access any memory location on the system including those locations mapping the [illegible]. Associated with the IBV-11 are two registers—a control [illegible] register and a data register—at locations 1776015[illegible] and [illegible], respectively. The control and data lines of the IEEE-488 instrument bus can be manipulated and monitored using these two registers [illegible] OPE utility thus circumventing the [illegible] software.

The 13378 interface does respond appropriately when the [illegible] Clear control line is asserted by the IVB-11 bus [illegible] to further interact with the 13378 have failed. Whenever [illegible] is made to address the GPIB adapter as a talker or as a listener, the ER2 bit of the status register for the IVB-11 is set [illegible] active listener or command acceptor on the instrument bus." [illegible] IEEE-488 handshaking signals were monitored with [illegible] NRFD and NDAC signals remain in the high state at [illegible] on the bus must bring NDAC low in order to accept [illegible] These handshaking signals were traced [illegible] where they are generated under the [illegible] microprocessor.

Support personnel at Spectral Dynamics were [illegible] The sequence of commands issued by the [illegible] IVB-11) were confirmed to be correct...at least they were [illegible] address a device as a talker or listener. It was recommended [illegible] Spectral Dynamics that the unit be returned [illegible] evaluation and repair. The necessary information [illegible] passed on to the RADC staff as well as information regarding the problems with the unit.

6.1.3 SD-350 Support Software

In parallel with the development of I/O software and the [illegible] of hardware problems with the GPIB Adapter, application software was developed to provide a user with access [illegible]. Three separate programs have been implemented [illegible] capabilities to the computer system user. All three [illegible] similar in that they allow the user to interactively change the SD-350's status, select a file in which to store spectral data, determine the time data on which the power spectrum is to be computed, and form header information associated with the new spectral data file. The differences between the three programs concern the source of the input data:

1. An analog signal applied to the analog input port of the SD-350
2. Digital data supplied via the host computer system

and the spectral analysis modes:

1. One single block of time data is converted to its power spectrum per analysis
2. Multiple time frames are analyzed and averaged to form a single output spectrum.

All source code for these programs and their associated subroutines reside on the REPT86 disk.

This software obviously hasn't been completely debugged given the hardware difficulties with the GPIB Adapter. The Status storage and display modules have been independently debugged. Also, there is an incomplete version of DSB available which outputs digital data to the SD-350 but does not obtain the resulting spectral information which is transferred with the IEEE-488 bus.

<u>SD-350 Subroutines</u> - Subroutines have been developed for the transmission of commands and data between the SD-350 and the host computer, and for processing/displaying this data:

1. STADIS—Displays current status of SD-350 parameters on terminal. The current state of the parameters are stored in the common STADAT, array PRES, and must be loaded into this data structure before calling STADIS. Figure 6.2 demonstrates the display presented to the user.

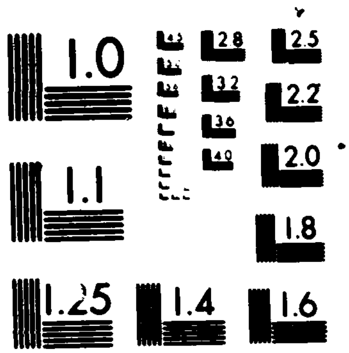
```
                 SD-350 STATUS
INPUT GROUP
    [IRA] RANGE (dB/1Vrms)             0dB
    [PFG] POST FILTER GAIN            20dB
    [HLD] HOLD                         OFF
ANALYSIS GROUP
    [TFS] TRANSFORM SIZE              1024
    [ARA] RANGE                          200.Hz
    [EXM] EXPAND MODE                 OFF/EXT
    [%RA] % RANGE                       5.0%
    [MAG] MAGNIFY                     EXT
AUX MEMORY GROUP
    [AMS] AUX MEMORY SOURCE           AVGR
    [AMT] AUX MEMORY TRANSFER         LOAD
AVERAGE GROUP
    [AVG] AVERAGE                     STOP
    [AVM] AVERAGE MODE                PEAK
    [#AV] NUMBER OF AVERAGES          128

TO CHANGE A PARAMETER:
   ENTER 'MNEMONIC, SPACE, NEW VALUE'
OR <CR> TO EXIT
>
```

FIGURE 6.2
DISPLAY OF SD-350 STATUS

AD-A176 129 RESEARCH AND DEVELOPMENT FOR DIGITAL VOICE PROCESSING 2/2
(U) ARCON CORP WALTHAM MA J D TARDELLI ET AL. OCT 86
R86-01W RADC-TR-86-171 F19628-84-C-0024

UNCLASSIFIED F/G 17/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

2. STAALT—Displays current status of SD-350 by calling STADIS and prompts the user for changes of parameters. Any changes are reflected in the PRES data structure and are transmitted to the SD-350 by calls to the subroutine CHGPAR.

3. STAGET—Obtain current status of all SD-350 parameters by calls to the subroutine GETPAR and updates PRES data structure.

4. GETPAR—Using QIO system directives directed to the IEEE-488 bus driver (device mnemonic IE:), this subroutine sends out the IEEE-488 bus commands directed to the SD-350's bus interface as required to read the current status of a particular SD-350 parameter.

5. PUTPAR—Similar to GETPAR except this subroutine writes a new parameter to the SD-350.

6. CHGPAR—Uses a combination of PUTPAR and GETPAR to change a parameter.

7. GETSPC—Sends instructions to the SD-350 via the IEEE-488 bus to transmit a specified number of bytes which represent spectral data. The number of bytes transmitted is a function of the transform size parameter. This data is read by the host and stored in a buffer array.

8. CONVDB—Converts the spectral data received from the SD-350 from a 12 bit integer format (2 bytes per spectral value) to a floating point number which represents a dB value relative to the full scale spectral value.

9. SD350—Controls the digital data transmission to the SD-350 from the host computer via the parallel interface developed from a DR-11C and additional digital logic. With this subroutine, the SD-350 can be placed in an "external data mode" wherein its internal A/D is disabled and it accepts 10 bit digital data. The 350 can also be placed in a "hold" mode in which the input memory of the SD-350 is held constant.

Digital Input Analysis DSB - The source code for this task is in the file DSB.FTN on the REPT86.disk. Digital time data is read from a specified file from the SPEECH DATA BASE and transferred to the SD-350 at a sample frequency of 8 kHz which is determined simply by software execution timing as there is no programmable clock on the system to provide specified interrupt intervals. The size of the transform and other control parameters may be specified by the user. Two user modes are available. In "manual" mode, the user specifies a starting time sample in the file which delimits the beginning of the current analysis frame. Once that analysis is complete and the resulting spectral data has been placed in an output file (if desired), the user is prompted for another starting point or is given the option to simply advance the starting point by a fixed number of time samples and form that spectrum. Manual mode is terminated by the user specifying no more analyses are desired. In "automatic" mode, the user is prompted for a starting point, the number of samples to advance each frame, the number of total spectrum to obtain, and the delay time between each spectral analysis. During program execution; frames of time data are sent to the SD-350, the resulting spectral data is written to a file (if desired), the spectrum is held on the display screen for the indicated delay period (specified in 60th's of a second). This process is reiterated without user intervention. Automatic mode is terminated when the specified number of spectra have been read or the end of the input file has been read.

Analog Input / Single Block Analysis ASB - The source code for this program is in ASB.FTN on the REPT86 disk. A single frame of time data taken from an analog signal can be selected by the user for spectral analysis. This selection is made by a keypress at the user's terminal. The Average Mode parameter (START,STOP,RESET) on the SD-350 can be used as a flag to indicate the completion of the spectral transform in the following manner:

1. the number of averages is always set to 1
2. on keypress, the averaging process is started
3. the software monitors the Average Mode parameter and loops until the averager is in the STOP mode meaning the "average" of the single transform is completed
4. the spectral data can then be read out of the averager's memory by the host computer.

Once the spectral data is stored in a file (if desired), the user has the option of selecting another time frame for analysis or terminating the program.

Analog Input / Continuous Mode Analysis ACM - The source code for this program is in ACM.FTN on the REPT86 disk. This program also takes time data from an analog signal applied to the input port of the SD-350. Unlike the single block mode program, this program provides a means for taking multiple spectrum (averaged or unaveraged) from a given input signal without the need for user intervention between each analysis. A time delay between each analysis frame can be specified. All of the resulting spectrum can be saved to disk.

6.2 SPEECH PROCESSING PERIPHERALS INTERFACING

The ADAMS-RUSSELL SPEECH PROCESSOR PERIPHERAL (SPP) provides Linear Predictive Coding of speech waveforms (Ref. 6.4). As a peripheral to a host computer, it provides a real time source of speech parameters for research purposes. Software has been developed to provide (1) communications between the PDP-11 and the SPP (2) real-time disk storage and retrieval of speech data associated with the SPP, (3) and conversion of LPC data between the format used by the ILS routines on the system and the format used by the SPP.

Communication between the SPP and the PDP-11 is by way of an asynchronous serial interface using the standard RS-232 protocol. The RSX-11M's terminal driver provides I/O software support on the PDP-11 end. Any of the RS-232 channels available on the PDP-11 for the purpose of interfacing with a terminal can be used to communicate with the SPP. A note of warning is advised concerning the use of the Clear-to-send line. The SPP internally pulls this line to + 5 volts. However, the DZ-11 pulls this line low thereby disabling the SPP's ability to write to the PDP-11. A solution is to simply disconnect this line on the RS-232 cable. This will not effect the channel's subsequent use with a VT-100 terminal.

The principle software commands used to communicate with the SPP consist of WTQIO system directives to read and write data to a LUN assigned to "TT4:". If some other terminal channel is to be used in the future, this assignment can be easily changed. The I/O function IO.RPR (read after prompt) has proven to be useful when a command is sent to the SPP with the expectation that an immediate response will be available for reading by the host.

Two programs are available for real-time communications with the SPP while maintaining disk storage/retrieval of LPC data. The source material for this software is on the REPT86 disk. The program SPPIN takes in analysis data for a specified period of time while packing the data into 512 byte blocks for writing to a disk file. After initialization, the SPP continually provides a new frame of data when it becomes available. Each frame represents 9 bytes of data—one header byte and eight data bytes. The SPP is capable of buffering the data upon reception of XOFF from the host, freeing the host to process one complete block of data. A double buffering technique was not required to maintain real time communications. QIOLIB library routines and QIO directives are used for disk I/O.

Conversion of SPP analysis data into an ILS compatible format is an option provided in the SPPIN program. The subroutine ILSFIL.FTN sequences through the SPP analysis data previously stored on disk and creates an ILS analysis vector for each frame. A separate file is created to hold the ILS formatted data. The subroutines MAKTAB.FTN, UNPACK.FTN, and DECODE.MAC are responsible for making the decoding tables and converting the SPP data to the ILS analysis vector format.

The corresponding synthesis program is called SPPOU. A specified disk file is used as the source of speech synthesis data which has been packed in the manner described for SPPIN. The SPP synthesizer is placed in "PLAY" mode wherein, the SPP transmits a frame request (FREQ) character whenever it is ready to synthesize another frame (158 sample points @130 microseconds per sample). The SPPOU program continually polls for a FREQ character and writes a frame of data to the SPP upon reception of this request. The user specifies the number of frames to output. A continuous loop is made through the specified output data until the user enters a Ctrl-Z at the terminal.

Conversion of ILS analysis data into SPP synthesis data is available by way of the CNVOUT program. Coding tables are created and the conversion of data is performed by the subroutines MAKCOD.FTN, PACK.MAC, and CODE.FTN. A check of the analysis file's header is made to insure that it is indeed an ILS analysis file. It also checks the original sampling frequency for compatibility with the 7692 Hz frequency expected by the SPP. An indication of any other sampling frequency results in a warning to the user. The resulting SPP synthesis data is temporarily stored in a disk file in the same format used by SPPIN and SPPOU. When all requested frames have been converted, the subroutine SYNOUT.FTN outputs the converted data to the SPP synthesizer in the same manner as SPPOU.

A user interface to this software is available at the login command file for the account RADC/SPP. Executable task image files are available at DR:[1,54].

These programs only begin to take advantage of the flexibility offered by the SPP in terms of the programmable options available. On startup, the LPC analysis and synthesis routines on the SPP use a standard set of parameters which provides LPC coding as defined by the Lincoln Lab LPC-10 algorithm. However, many of these parameters can be altered via commands sent over the RS232 connection. Table 6.1 lists these programmable options. The LPC parameter quantization and coding schemes are also programmable as the number of bits allocated to the various parameters and the coding/decoding tables are user definable. The SPP software does not presently provide the capability to control these options. They could be provided, as needed, in the future.

Table 6.1
LIST OF PROGRAMMABLE OPTIONS FOR SPP

Analyzer / Pitch Detector Parameters

- -Number of samples per frame
- -Hamming window size
- -Order of Linear Predictive Analysis (.LE. 15)
- -Digital pre-emphasis filter coefficients
- -Correlator input downscaling factor
- -Energy estimate (residual energy vs. input signal energy)
- -Average pitch period limits used by Gold pitch detector
- -Maximum allowable pitch period
- -Pitch detector's input low-pass filter coefficients
- -Silence threshold

Synthesizer Parameters

- -Order of synthesizer filter (.LE. 15)
- -Numbers of samples per frame
- -Reflection coefficient interpolation (linear) frequency
- -Reflection coefficient interpolation slope
- -Digital de-emphasis filter coefficient
- -Energy estimate (residual energy vs. input signal energy)

Table 6.1
LIST OF PROGRAMMABLE OPTIONS FOR SPP

6.3 INTERACTIVE LABORATORY SYSTEM

The Interactive Laboratory System (ILS) is a modular software package for computer use in research involving sampled data and signal processing (Refs. 6.5, 6.6). It has been programmed to operate in an interactive, multiuser mode. The ILS package distributed by Signal Technology, Inc. consists of about 90 main programs and about 250 Fortran subroutines. There are seven assembly language subroutines provided for the primitive disk I/O operations.

Specific tasks for processing and analyzing data of interest are performed by sequentially invoking ILS task modules. Each module is a single ILS "program" which can be executed by the user via an MCR request consisting of a three letter mnemonic plus any accompanying command values (this assumes, of course, that the ILS module is "installed"). Thus the user interface can be quick and efficient once a familiarity with the mnemonics and command alternatives is obtained. A more user-friendly interface to the ILS package has been developed on the RADC system and will be discussed later in this section. A provision for communication of parameter values between program modules is made possible by providing, on disk, an exclusive file for each user—the COMMON file. The COMMON file contains global system parameters and it serves as a work area for deposit and retrieval of information by all commands executed by the user. In this way an ILS module can operate on previous results and arguments passed through the user's COMMON by a preceding module.

Because of the modularity of the system, any program module may be modified without affecting the other modules. This feature also permits the replacement or addition of program modules on disk providing they are properly designed to be compatible with the ILS conventions. Within each ILS program another level of modularity is seen in which all programs are composed of subroutine and function calls to standardized segments of code residing in a well designed library. Thus the ILS system is very amenable to custom alterations of signal processing and analysis software.

6.3.1 ILS Installation

ILS version 3.0 source code was read from a 9-track magtape onto a RL02 disk using the VAX/VMS system at Arcon Corp. The VMS utility "EXCHANGE" was used to convert the DOS-11 source files to FILES-11 format. The ILS object library and tasks were built on the virtual disk labeled ILS. The software was built while operating under UIC [1,1]. A command file ILSASN.CMD is available to make the required logical device assignments for the build process. The contents of the important UIC's on ILS are:

* [100,300] contains the object library ILSLB.OLB .
* [100,302] contains source files.
* [100,305] contains the build command files.
* [100,306] contains the task files.

All of the source code was compiled with the Fortran IV compiler which was installed as ...F4P to be compatible with the command files used to direct the software installation. The library objects and final tasks were all built with the command files obtained from the magtape except for three tasks (TFU, SDI, and SIF) which did not build properly under direction of these command files. The problem involved the overflowing of the 16-bit virtual address space available on the PDP-11. A solution to the problem was to overlay these tasks. The overlay descriptive language (ODL) files for TFU and SDI can be found at ILS:[1,1] while SIF was rebuilt with no trace-back enabled.

Additional tasks have been added to the ILS package for specific research purposes as required by the work going on at the Speech Lab. In particular, two tasks have been added which provide LPC analysis and synthesis of speech signals using some of the algorithm enhancements offered by G. Kang and S. Everett of NRL. These tasks, KEA and KES, were developed on the virtual disk KANG and are fully described in Chapter 5 of this report. In terms of using these tasks, the KEA algorithm takes the same input parameters as the ILS task API while KES is just a revision of the SNS task provided by ILS. The phonetic onset detection algorithm devised by Kang and Everett has also been incorporated into the ILS package as a subroutine in the DSP task used to display time signals at a graphics terminal. This option is selected by using an "O" alphabetic parameter in the DSP run command string. The source code for ONSET.FTN can be found on the ILS disk.

After approximately 6 months of user experience with the ILS software package, a number of problems and inconveniences were recognized. Work was done to provide a more "user-friendly" user interface to the package as well as providing solutions to known problems and inadequacies that researchers have encountered. The retention of as much of the original ILS code as possible was always a consideration during this work.

6.3.2 ILS Improvements

ILS Menu Interface - A major inconvenience, especially to the researcher who uses the package sporadically, is the multitude of 3 letter mnemonics required to specify particular ILS functions. It is difficult to remember which mnemonic goes with which function. A new user-interface to the package was created using the indirect command file facilities available on RSX. This interface is in the form of a central menu and a set of sub-menus all contained in the indirect command file ILS.CMD. At the sub-menu level, the user is presented with a display of a group of ILS task mnemonics and a brief description of each function. A prompt for the function to be executed is made to which the user enters the function's mnemonic and an appropriate command line. At any point, the user can call the ILS help task, HHH, for a description of the format of a particular function's command line. A response of a simple carriage return to the menu's prompt allows the user to leave the current menu. Table 6.2 illustrates an example of a sub-menu display from ILS.CMD.

```
>;                             FILE UTILITIES
>;ALL FILES
>;  FIL - Create / Specify / Delete
>;  PRT - Print Data
>;  HDS - Display Speech Data Base Header Info.
>;  HCR - Initialize Speech Data Base Header
>;  HMD - Modify Speech Data Base Header
>;DATA FILES
>;  INA - Initialize Analysis Parameters
>;  TRF - Transfer Data Frames
>;  TTL - Transfer Marked Data and Lables
>;  VER - Verify Header Blocks
>;RECORD FILES
>;  LRE - List Records + Headers
>;  OPN - Allocate / Open
>;  SRE - Generate Record File
>;  TRE - Transfer Records to Secondary File
>;LABEL FILES
>;  LBA - Label Data Segment
>;  LBF - Select / Create Label File
>;  LLA - List
>;  TLA - Copy Labels to Secondary File
>;<CR>- EXIT
>* ENTER MNEMONIC [SPACE] COMMAND [S]:
```

TABLE 6.2
AN ILS.CMD SUB-MENU

Dynamic Installation Of ILS Tasks - A system level problem with the package is the fact that not all of the ILS tasks available can be installed at one time which would allow any user to execute any task by simply typing its mnemonic and command line. The limitation on the number of installed tasks is related to the finite amount of system "pool memory" available for such things as holding task control block information for each installed task. Complicating matters more, a user must be privileged in order to install a task in RSX. If this were not the case, the command file interface to ILS could have been written such that a specified task for execution could be installed before execution and then removed upon completion.

In response to this problem, a pair of programs were developed which provide the means for any user to request that ILS tasks be dynamically installed and removed as needed. The source code for these tasks can be found on the REPT86 disk. The DYNSRV task manages the actual installation and removal of ILS tasks. DYNSRV must be installed (TASK=...srv) and executed from a privileged terminal and must remain running at any time requests for ILS tasks might be made. However, DYNSRV does not vie for system resources except upon reception of a request for action. (It should be mentioned that DYNSRV does occupy memory at all times; however this is not a scarce resource on the system at this time). DYNSRV maintains a table of 5 installed ILS tasks. If a requested task is not found in the table indicating that it is already installed, a task is swapped out of the table and removed from the system on a "least-recently-used" basis. The request to DYNSRV for this service is made by another task, called ILSREQ and installed as ...IRQ, executed by the user. The intertask communication capabilities of RSX are used for making the requests. Status is returned to IRQ concerning the success of the installation of the requested task. Following successful execution of IRQ, the user can execute the desired ILS task just as he would any other installed task. The user who interacts with the ILS software package by way of the ILS.CMD menu interface is unaware of the need to request the installation of tasks as IRQ is called within the command file.

The STARTUP.CMD file for the system was edited such that DYNSRV and ILSREQ are installed at startup. Also, the virtual disk ILS is mounted as a public device and assigned the logical IL: as it serves as the source of ILS task files. The DYNSRV task is executed during the indirect processing of the STARTUP.CMD file.

ILS Filename Specifications - An inconvenience recognized by users of the package involves the specification of filenames for the ILS "primary" and "secondary" files used for input and output by ILS tasks. The convention is to name all files with the format of WDNNNN. where NNNN is a numerical value. Thus, the user only needs to specify a file number and the corresponding filename is created. However, it would be nice to be able to work with files having names with formats other than the WDNNNN. convention (e.g., data files created by tasks using the QIOLIB routines).

The ILS task FIL enables a user to specify the primary and secondary files to be used by subsequently invoked ILS tasks. FIL was revised in order to allow the user to specify files with any filename. For specifications [illegible] than those using the conventional WDNNNN. format, the use[illegible] a negative file number to FIL. A positive file number [illegible]ult in the creation of a conventional filename as in the past. However, a negative file number results in a prompt for a filename which the user provides. Note that only the filename and extension are prompted for at this point and not the device and UIC. The directory pathname to be used is established by the ILS task TBL and the second numerical argument presented to FIL. The formation of the new filename is done in a new subroutine GETNAME which is called if the file number provided by the user to FIL is negative. The subroutine CHKFL which is used by all ILS tasks was revised so that if the current primary or secondary file number is negative, a new file name is not created using the default, conventional format but instead, the previously user-supplied filename stored in the common file by FIL is retained for use.

ILS Directory Utility - If the user does elect to stay with the WDNNNN filename convention, a utility was written which will provide a directory listing of all files on a specified device and UIC with this filename format. This listing includes the 14 character string of text placed in the file header providing a "title" to the data file. Also listed is the creation date of the file and its ILS file number. The source code and task build command files are IL:[7,161]GMCDIR.FTN and GMCDIR.CMD, respectively.

The task file for this utility is installed at startup and can be invoked by the MCR command DIR. The command line can include a specific device mnemonic and UIC in the normal RSX-11 format (e.g., DR:[200,200]). If no device or UIC specifications are given in the command, the default SY: device and default UIC are used. The default UIC determination is made using the GETTSK (i.e., get task control block parameters) system directive. Much of the coding for this utility is concerned with parsing the input command line in attempts to make it fault-tolerant. For example, if the "[" signs are left off the UIC specification a successful parse is still possible.

Once the device and UIC to be searched is determined, the directory file on the specified device's [0,0] UIC for the specified UIC is opened. All entries in this directory are first compared with the template WDNNNN. (i.e., for the first two characters of the filename = WD, and a null extension). On a successful match, the file pointed to by the entry is opened and the date and title fields of its header are read. The values in these fields are displayed on the terminal. Should no file be found with this filename convention, a message is displayed saying "No ILS Data Files Found".

Radix Of ILS Unit Numbers - A problem was discovered with the ILS tasks which occurred only if the user operates from a virtual disk (VD) assigned to a VD unit number greater than or equal to 8. The ILS subroutine GTDEF determines the user's current SY: device and unit number as required for specifying the complete filename of the user's common file (WD9999.) which is accessed at the beginning of all ILS tasks. The SY: device's unit number is converted to ASCII characters using decimal notation. However, the driver for the virtual disks expects the unit numbers to be specified in octal notation. Consequently, the common file is never found if the user's SY: device has a unit number greater than 7.

The solution to this problem consists of including a new subroutine called OCT2AS in the ILS library which converts a binary value into ASCII using octal notation. This subroutine is now called in GTDEF and replaces the original call to the subroutine I2AS. OCT2AS is a simple assembly language program which makes a call to the RSX system library routine $CBOMG for the conversion to ASCII.

6.4 SPEECH DATA BASE

A speech data base is available for use at the RADC/EEV speech processing facility (Ref 6.7). This data base, along with accompanying software, enables users to store and manipulate data from any of the various processors at the speech lab. All speech data base files are created and manipulated by the routines found in QIOLIB. Each file has a one-block (512 bytes) header record which indicates the format and origin of its content. This header format has been slightly altered to be congruent with the data file format of ILS (see the accompanying section on ILS). Data is stored in direct access, block I/O files, in either 16-bit integer or 32-bit floating point formats. The 16-bit integer format is the same as the ILS "sampled data file" format; however, there is no provision in ILS for a floating point sample data file.

The most significant change to the speech data base files involves the header value used to record the number of data blocks in the file. ILS records this value in terms of blocks containing 64 samples each as compared to the convention used in the Speech Data Base in which each block represents 256 words of data. The new header format follows the ILS convention with 64 words/block quantization. For a task using the QIOLIB routines to access one of these files, the value passed into a common used to hold header information is in terms of the 256 words/block convention. This conversion is done in the QIOLIB subroutines RHQIO and WHQIO which transfer data between the file header and the common. Thus, the first approach consisted of dividing by 4 the value read from the file by RHQIO before storing it in the common, and multiplying by 4 the value written to the file from the common by WHQIO. This technique works fine until the number of 256-word blocks in a speech data base file becomes greater than (2**15)/4 = 8192. Upon multiplying such a number by 4, the product becomes negative since 16-bit 2's complement representation is used. Thus when the value is read out later, and divided by 4 the indicated number of blocks available in the file is negative! This causes problems, obviously, if the program reading this value out of the file header uses it for program control as is the case in the MAPOUT program. At an 8 kHz sampling rate, only 262 seconds of continuous digital recording can be handled before this problem presents itself. This length of time is not adequate if the recording of a complete DRT list is wanted.

A solution to this problem was found that at least allows the storage of 524 seconds of audio information in one file and the retention of the current file header format. The number of 256-word blocks in a file is still multiplied by 4 upon writing out the file header. However, in place of dividing the number by 4 upon a subsequent read, two logical right shifts are done on the value to determine the number of 256-word blocks in the file. This operation guarantees that the resulting value is always positive (i.e., the MSB is always a 0). The maximum number of blocks in the file is now (2**16)/4 = 16384 since an overflow will occur should a value greater than this be multiplied by 4 and forced into a 16-bit integer representation.

Many of the header values were relocated in the header block because their previous locations conflict with usage of these same locations by the ILS software. A mapping of the new values in the header block as used by the Speech Data Base is given in Table 6.3. It should be noted that the ILS programming guide states that the header locations 32-57 are unused by the ILS routines. Also, the default header size for the ILS files has been set to 256 words in order to be compatible with the Speech Data Base even though the ILS software will not use any of the locations beyond 64.

TABLE 6.3
FILE HEADER BLOCK FORMAT SPEECH DB AND ILS DATA FILES (JUNE 1985)

Variable types: A - ASCII, F - Fl. Pt., I - Integer, R - RAD50
Lengths given in words. '*' indicates speech data base variable.

NAME	START	LENGTH	TYPE	CONTENTS
N (IFRAME*)	1	1	I	NUMBER OF POINTS PER ANAL. WINDOW
M	2	1	I	NUMBER OF AUTOREGRESSIVE COEFFS.
ICON	3	1	I	PREEMPHASIS CONSTANT (0-100)
NSHFT	4	1	I	SHIFT INTERVAL PER DATA FRAME
IHAM	5	1	I	HAMMING WINDOW ('Y' OR 'N')
NSPBK (NBLKS*)	6	1	I	# BLOCKS IN FILE (NOT INCL HDR)
NP	7	1	I	NUMBER OF RESONANCE PEAKS
ISTAN	8	1	I	STARTING FRAME FOR ANALYSIS
NAN	9	1	I	NUMBER OF FRAMES ANALYZED
NFR	10	1	I	# OF VAR. SIZED FRAMES ANALYZED

TABLE 6.3 (Continued)
FILE HEADER BLOCK FORMAT SPEECH DB AND ILS DATA FILES (JUNE 1985)

NAME	START	LENGTH	TYPE	CONTENTS
MU	11	1	I	# OF AUTOREGRESSIVE COEFFS.
NT	12	1	I	DOWN-SAMPLING FACTOR
IFLD(1)	13	1	A	FIELD 1 - 2 ALPHABETICS
IFLD(2)	14	1	A	FIELD 2 - 2 ALPHABETICS
IFLD(3)	15	1	A	FIELD 3 - 2 ALPHABETICS
IFLD(4)	16	1	A	FIELD 4 - 2 ALPHABETICS
NSC	17	1	I	STARTING SECTOR FOR ANALYSIS
IAFIX	18	1	I	FLAG FOR AUTOREGRESSIVE COEFF.
IDK	19	1	I	DISK # OF DATA FILE ANALYZED
NFL	20	1	I	FILE # " " " "
ID (1)	23	1	A	IDENTIFICATION
ID (2)	24	1	A	IDENTIFICATION
ID (3)	25	1	A	IDENTIFICATION
ID (4)	26	1	A	IDENTIFICATION
ID (5)	27	1	A	IDENTIFICATION
NASC	28	1	I	NEXT AVAILABLE SECTOR (TTL)
NAPT	29	1	I	NEXT AVAILABLE POINT (TTL)
NZERO	30	1	I	NUMBER OF ZEROS (TTL)
FLAG	31	1	I	1111 IF SECONDARY FILE INITIALIZED
ITYPE*	32	1	I	DATA TYPE: RAW = 0, PROCESSED =1
IFRMAT*	33	1	I	DATA FORMAT: FLT PT =0, INTGR =1
IRLCX*	34	1	I	DATA FORMAT: REAL =0, CMPLX =1
IDEV*	35	1	I	SOURCE PROCESSOR (INTGR CODES)
ENGPW*	36	2	F	ENERGY/POWER
SGNOI*	38	2	F	SIGNAL/NOISE RATIO
RAWFIL*	40	5		SOURCE RAW FILE NAME - FORMAT:
IRFLNM(3)*			R	3 WDS. - RAD50 FILE NAME
IRFLEX*			R	1 WD. - RAD50 EXTENSION
IRFLVR*			I	1 WD. - INTEGER VERSION #
IALG*	45	1	I	COMPRESS. ALGORITHM (INTGR CODES)
ISNCOM*	46	1	I	SYNTHESIZED = 0, COMPRESSED = 1
IWINDW*	47	1	I	WINDOW FUNCTS. USED (INTGR CODES)
IHFLT*	48	1	I	FILTER - HI LIMIT
ILFLT*	49	1	I	FILTER - LO LIMIT
ICHAN	58	1	I	STARTING A/D CHANNEL
NCHAN	59	1	I	NUMBER OF CHANNELS
MULAW	60	1	I	SET TO 50 IF 8-BIT LOG QUANT.
IPWR*	61	1	I	POWER OF MULT. FOR SAMPLING FREQ.
IFRQ*	62	1	I	SAMPLING FREQ.
FLAG	63	1	I	-32000 = SAMPLED DATA FILE -30000 = RECORD DATA -29000 = ANALYSIS DATA
XXXXX	64	1	I	32149
NSIZ1*	65	1	I	-
NSIZ2*	66	1	I	-
MAXCUR*	67	1	I	CURRENT MAXIMUM AMPLITUDE
MAXORG*	68	1	I	ORIGINAL MAXIMUM AMPLITUDE
DATE*	100	5	A	CREATE DATE - (DD-MMM-YY)
TITLE*	105	10	A	DESCRIPTIVE TITLE - 20 CHARS MAX
COMMNT*	115	142	A	COMMENTS

6.5 PRECISION FILTER INTERFACING

Remote control of the Precision programmable filters Model 636 used for anti-aliasing purposes in the Speech Lab is now available. Gain and cutoff frequency values can be specified for each of the 10 filter channels currently available in the unit (channels 0-7, and 14,15). Data is sent to the filter unit in the form of ASCII bytes in serial fashion via standard RS-232 transmission lines. Interfacing at the filter unit's end is provided by a "Model 636-C-02 Interface card", while a DL-11 asynchronous serial interface card (system device TT1:) is used to interface into the PDP-11's Unibus. The TT1: serial channel is shared for use by both the filters and the Quintrell processors with multiplexing done by way of the manual switch box located above the Quintrells.

This new communications channel provided by the DL-11 connected to the filter unit can be viewed as an additional "terminal" added to the system which can only receive messages of a very limited vocabulary. From an application's software point of view, this means that the parameters of particular filter channels can be altered at run time by simple WRITE statements directed to a Logical Unit Number (LUN) assigned to this "terminal". Furthermore, these parameters can be altered by a user seated at one of the computer terminals in an interactive manner. That is, a person can simply send a command to the "filter terminal" using the RSX-11M utility called BROADCAST (BRO); no actual programming is required.

The filter unit must be powered up, the Remote switch must be in the upper position, and the switch box above the Quintrell processors must direct the input from the PDP-11 to the "EIA" channel in order to transfer commands from the computer to the filters. The following is a brief description of the possible commands:

1. "C" followed by two decimal digits to select a filter channel
   Example— "C07" selects channel 7.
2. "G" followed by a "1","2","3","4" to set gain of selected channel.
3. "F" followed by four digits which sets the mantissa of the cutoff frequency. Note that there is always an implied decimal point following the second digit. The mantissa can range from 00.01 to 10.23.
4. "E" followed by the sign of the exponent (always "+") and a single digit which specifies the exponent of the cutoff frequency.

For additional details on these commands, consult Ref. 6.8. An example of a command sent via BRO would be:

```
>BRO TT1:C02G3F0900E+2
```

which would set the gain of channel 2 to level 3 and the cutoff frequency to 900 Hz.

CHAPTER 7

COMMUNICABILITY AND VOCODER SYSTEMS

7.1 COMMUNICABILITY TEST CONFIGURATION

The communicability facility at the RADC/EEV Speech Processing Lab is designed as a half-duplex system (see Figure 7.1). The overall system is designed for versatility. The facility has the capability to allow for changing from speech processor to speech processor. The system also features plug-in headsets which can be interchanged quickly. This versatility adds an extra dimension to the system; the ability to test many different harwares with minor configuration changes.

Users communicate on a push-to-talk basis. Both test stations reside in acoustic isolation rooms. Speech from either station is channeled to the test administrator's room. Here a control box and relay system (see Figure 7.2) routes the signal to its appropriate paths, sending unprocessed sidetone to the speaker of the moment, and also sending the signal to the computer room to be processed and eventually to arrive at the other station. A Vu meter, calibrated to read 0Vu when seeing 5Vp-p, monitors the send to the computer room. This allows the test administrator to set the line amplifier level so as to not overload any processors.

All audio lines are shielded, balanced cable. Transformers at the patch panel in the computer room step this signal down to unbalanced and from here a host of signal processing devices may be accessed. Before leaving the computer room, the processed signal is transformed to a balanced state. The signal is then returned to the relay system where it is sent to the current listening position. A transformer resides at both listening positions. Its function is to step the signal down to low impedance unbalanced, to properly match the headsets.

The patch panel in the computer room allows access to a number of devices including the Quintrell signal processors, the MAP300 associated with the PDP 11/44, the FPS AP-120B,the PDP 11/34 and the MAP-300's that currently implement the DCEC algorithms, and any other processor that uses standard line levels and impedances. In addition a host of active and passive filters are available as well as an oscilloscope, a voltmeter, a counter and two Spectral Dynamics digital signal analyzers (SD350 and SD360).

The test administrator resides in Room 3. From here the test may be monitored. LEDs on the relay system indicate which position is sending a signal. The Vu meter reflects output to the processor. A headphone monitor on the line amplifier assures that an undistorted signal reaches the computer room. Another feature is the master microphone override. This allows the administrator to communicate in unprocessed speech to both stations simultaneously.

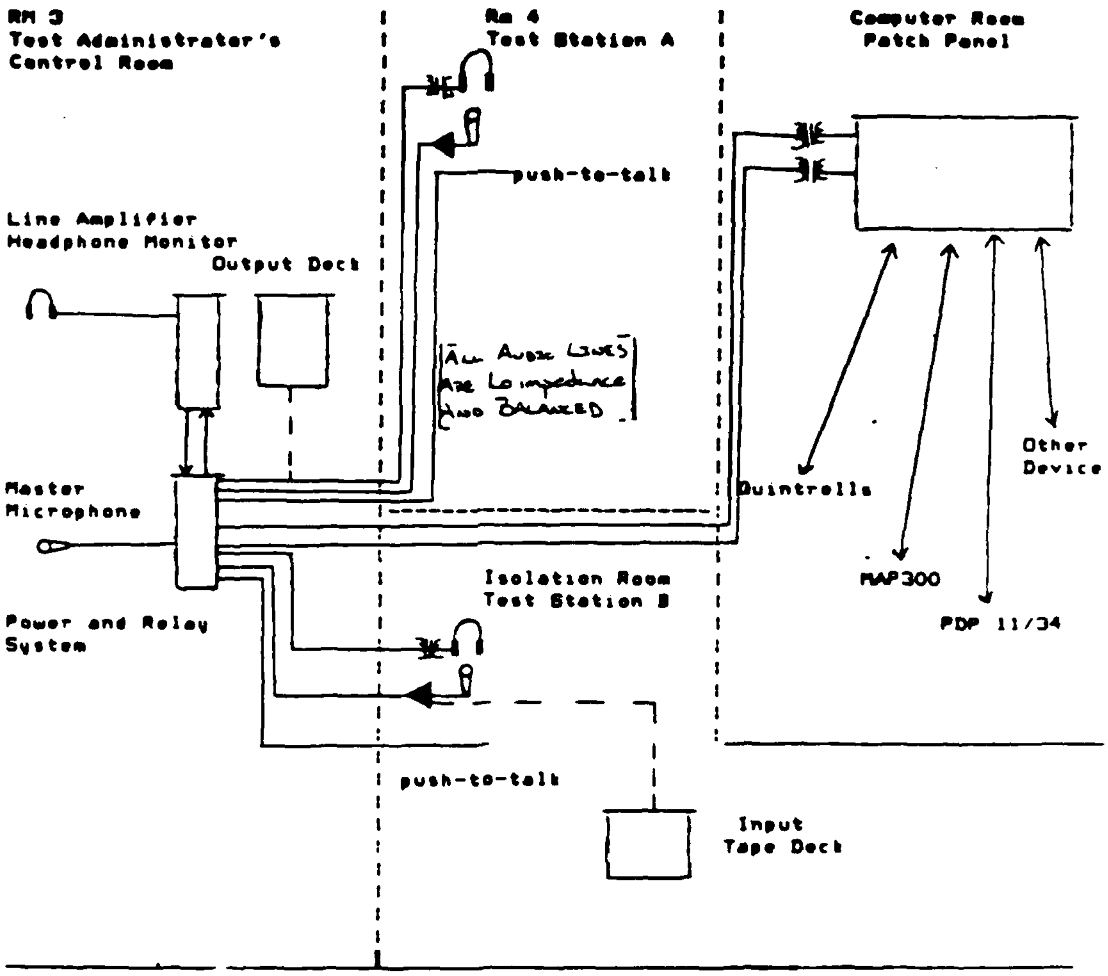


Figure 7.1 Communicability Facility

A final feature incorporated into the communicability test is the option of inserting tape recorders. With the use of a switch that effectively locks the signal flow so that station B is always sending and station A is always receiving, an input deck is placed at station B. Pre-recorded test material leaves station B and travels through the communicability system, including any chosen processor, after which it is accessed for recording at the output to station A on the relay box.

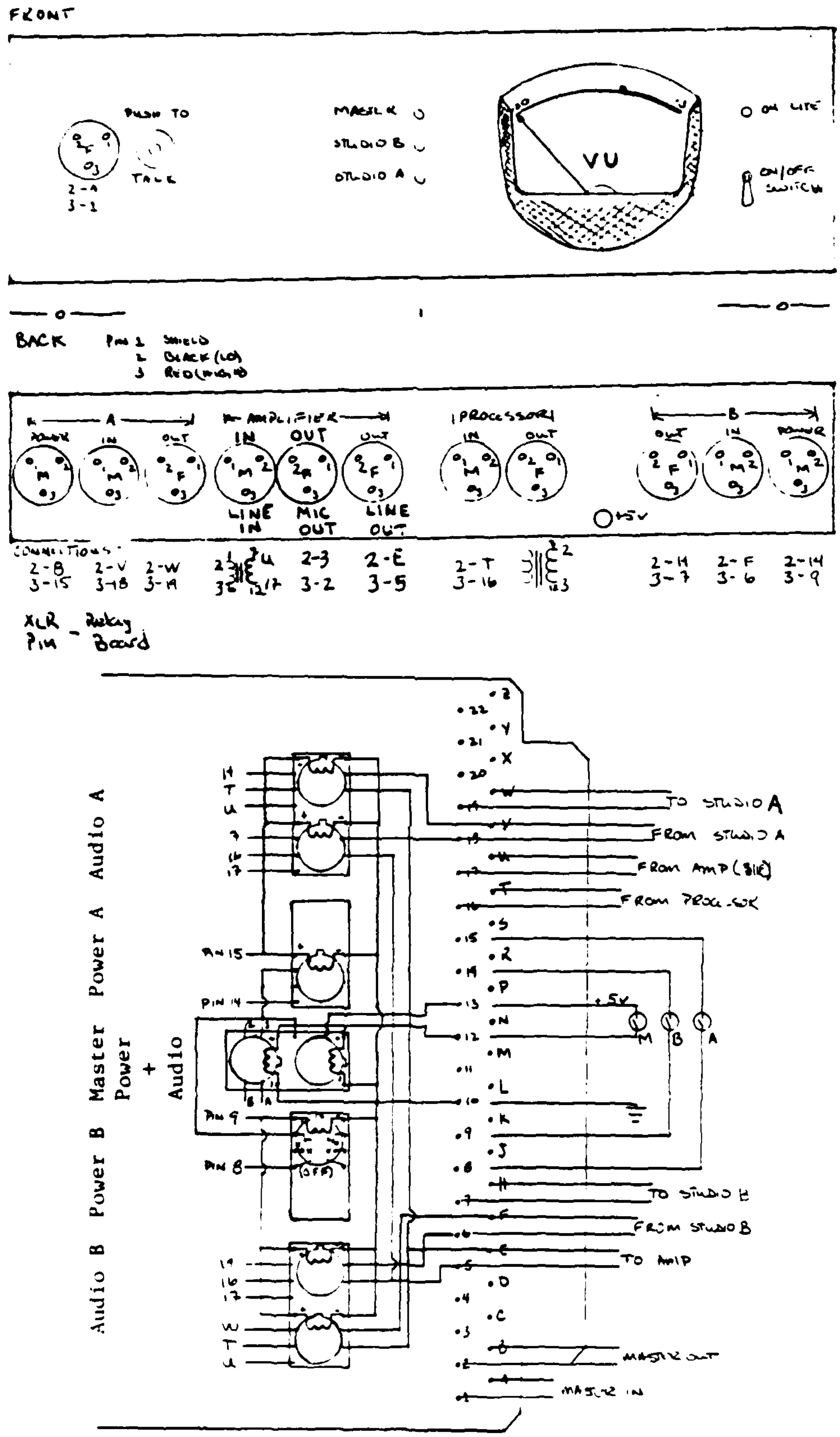
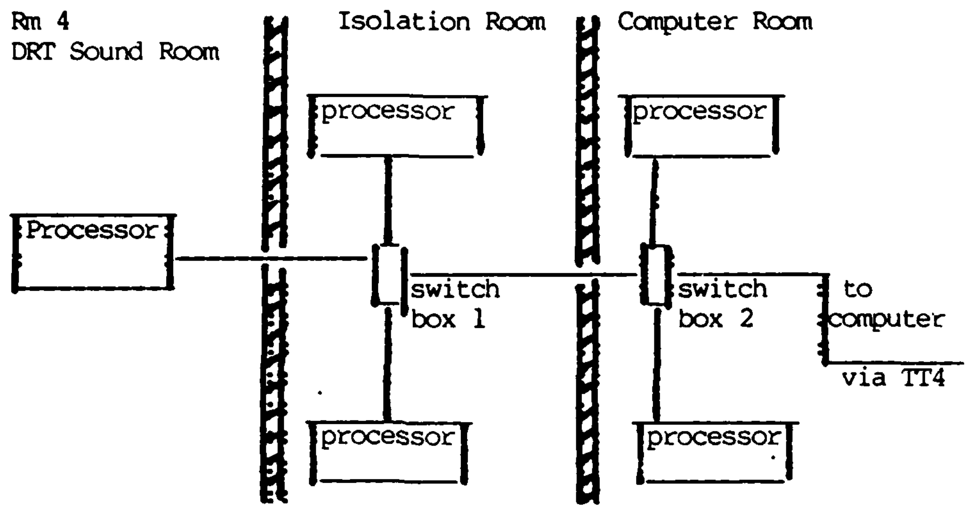


Figure 7.2
Communicability Control Box and Relay System

7.2 COMPUTER/VOCODER COMMUNICATION SYSTEM

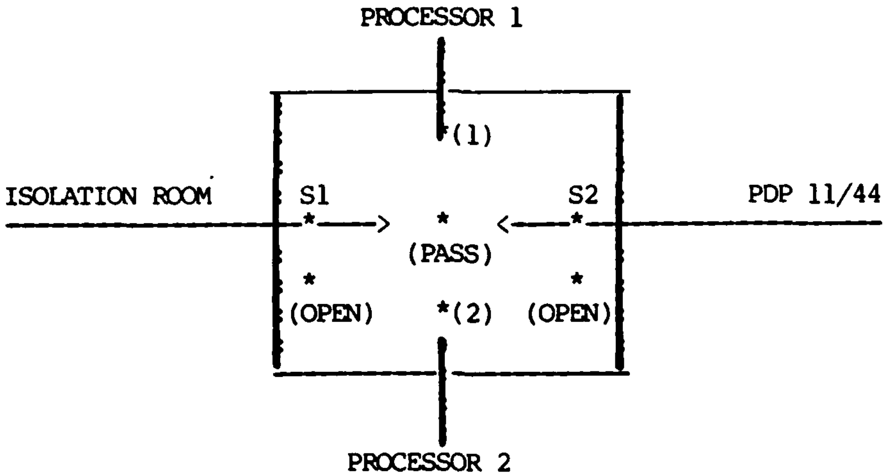
The purpose of this system is to test and demonstrate stand-alone vocoders and to allow interaction of these processors with the PDP 11/44. The system consists of 5 lead RS232 cables connecting the DRT Sound Room, Room 4, with the isolation room, connecting the isolation room with the computer room, and connecting the vocoder station in the computer room with the PDP 11/44 (see Figure 7.3). At each junction a switch box is present to allow a choice of processors to be accessed. When necessary, null-modem adapters are placed at the vocoder's RS-232 port. Only one combination of compatible processors may be operated at any one time. For instance, an operator in the computer room and an operator in the DRT Sound Room cannot both access a device in the isolation room. Once a path is chosen, vocoders are operated according to their own instructions.



Note - All lines are 5 lead RS-232

Figure 7.3
Computer/Vocoder Communication System

System User Instructions - The heart of the Computer/Vocoder system is in the switch boxes. These boxes direct signal flow to allow any processor to access any other station. They consist of two 5-pole, four position switches. Each switch directs the incoming lines. For instance, the box in the computer room has a switch which directs the line coming from the computer and has a second switch to direct the line coming from the isolation room. Either line can be pointed to either processor or the signal can pass straight through the box allowing direct communication between the computer and processors in the isolation room. The fourth position of each switch is simply open to allow disconnection of any particular path. With these switching combinations, any processor may access any other processor or the computer (see Figure 7.4).



Note - Both switches can access 1, 2, PASS, and OPEN

Figure 7.4
COMPUTER/VOCODER SWITCH BOX

REFERENCES

1.1 S. Woolf, et al., "Interactive Mathematical Modeling", Chapter IX, Speech Processing; RADC-TR-77-86 February 1977; ARCON Corporation, Final Technical Report 1 October 1973 to 1 September 1976.

1.2 W.H. Rosenfeld and C.B. Hayes, "Speech Processing Software and Computer Systems Development";R77-01W ARCON Corp., 4 Feb 1977; Final Report 1 September 1975 to 30 September 1976.

1.3 W. Rosenfeld, "Narrowband Digital Voice Processing (LDVT PLPC/TRIVOC Software Documentation) ESD TR-76-282, Vol. IV, March 1978.

1.4 J.D. Tardelli and C.B. Hayes, "Narrowband Digital Voice Processing Software and Computer System Development; Quintrell and PDP-11 Programs and Systems"; R78-01W ARCON Corp., 24 April 1978; Final Report 1 October 1976 to 31 December 1977.

1.5 J.D. Tardelli and C.B. Hayes, "Mathematical Modeling for Narrowband Digital Voice Processing - A Quintrell Implementation of AFSC LPC-10 and Voice Processing Facility Systems"; R80-01W ARCON Corp.,30 Sept 1980;Final Report 1 Jan 1978 to 30 April 1980.

* 1.6 C.M. Walter and J.D. Tardelli, "Array Processor Oriented Signal Compression Techniques for Optimizing the Performance of Narrowband Digital Transmission Systems"; R81-01W ARCON Corp., 30 Oct 1981;Final Report 1 May 1980 to 30 Sept 1981; and RADC-TR-85-41, March 1985.

1.7 E. Wachsler and J.D. Tardelli, "Speech Processing Facility System and DRT Software"; R82-01W ARCON Corporation, 20 April 1982; Final Report 1 May 1980 to 31 December 1981.

* 1.8 J.D. Tardelli and W.F. Bellew, "Diagnostic Rhyme Test Software System"; R83-01W ARCON Corporation, 31 January 1983; Interim Report 1 January 1982 to 31 December 1982; and RADC-TR-85-240, December 1985.

* 1.9 J.D. Tardelli, C.M. Walter, J.W. Kennedy and J.T. Sims, "Research and Development for Digital Voice Processing"; R84-01W ARCON Corporation, 31 July 1984; Final Report 1 January 1982 to 8 May 1984; and RADC-TR-85-46, March 1985.

1.10 J.T. Sims and J.D. Tardelli, "The Effects of Controlled Speech Level Input on the Intelligibility Testing of Speech Compression Algorithms" Proc. 1985 IEEE ICASSP, Tampa, FL, 1985, No. 11.15, pp 430 - 433.

1.11 J.D. Tardelli, J.T. Sims and P.D. Gatewood, "RADC/EEV Diagnostic Rhyme Test System Improvements"; R86-02W ARCON Corporation, To Be Published.

1.12 J.T. Sims, "A Speech-To-Noise Ratio Measurement Algorithm," J. Acoust. Soc. Am., Vol. 78, No. 5, pp. 1671-1674, 1985.

2.1 C.M. Walter and J.D. Tardelli, "Signal Compression Model Interrelationships in the Time, Frequency, Principal Component and Canonical Coordinate Domains"; Proc. 1982 IEEE ICASSP, Paris, France 1982, pp 1367 - 1370.

* 2.2 C.M. Walter and J.D. Tardelli, "Array Processor Oriented Signal Compression Techniques for Optimizing the Performance of Narrowband Digital Transmission Systems"; R81-01W ARCON Corp., 30 Oct 1981; Final Report 1 May 1980 to 30 Sept 1981; and RADC-TR-85-41, March 1985.

* 2.3 J.D. Tardelli, C.M. Walter, J.W. Kennedy and J.T. Sims, "Research and Development for Digital Voice Processing," R84-01W ARCON Corporation, 31 July 1984; Final Report 1 January 1982 - 8 May 1984; and RADC-TR-85-46, March 1985.

2.4 J.D. Tardelli and C.M. Walter, "A Speech Waveform Analysis and Reconstruction Process Based on Non-Euclidean Error Minimization and Matrix Array Processing Techniques"; Proc. 1986 IEEE ICASSP, Tokyo, Japan 1986, no. 24.4.

3.1 J.S. Lim and A.V. Oppenheim, "Enhancement and Bandwidth Compression of Noisy Speech," Proceedings of the IEEE, vol. 67, pp. 1586-1604 (December 1979).

3.2 "Instruction Manual: SD360 Digital Signal Processor (SN 191 and on)," Spectral Dynamics Corp., San Diego, CA, March 1979.

* 3.3 J.D. Tardelli, C.M. Walter, J.W. Kennedy and J.T. Sims, "Research and Development for Digital Voice Processing," R84-01W ARCON Corporation, 31 July 1984; Final Report 1 January 1982 - 8 May 1984; and RADC-TR-85-46, March 1985.

3.4 "General Information: SD350 Digital Signal Analyzer," Spectral Dynamics Corp., San Diego, CA, no date. Spectral Dynamics "SD350 Digital Signal Analyzer", March 1978.

3.5 S.M. Kay and S.L. Marple, "Spectrum Analysis—A Modern Perspective," IEEE Proceedings, vol. 69, pp. 1380-1419 (November 1981).

3.6 Digital Signal Processing Committee (IEEE ASSP Society), Programs for Digital Signal Processing, IEEE Press, New York, 1979.

3.7 C.F. Teacher, "AFSATCOM, LPC-10, Secure Voice Evaluations: Characterization of Aircraft Noise and Audio Systems in the EC-135 and E-4B Aircraft," KETRON, Inc., Wayne, PA, 1983; Final Report 22 April 1982 through 22 April 1983.

3.8 C.P. Smith, "Narrowband (LPC-10) Vocoder Performance Under Combined Effects of Random Bit Errors and Jet Aircraft Cabin Noise," RADC In-House Report No. RADC-TR-83-293, December 1983.

* 3.9 C.P. Smith, "An Evaluation of Intelligibility Performance of the DOD Standard LPC-10 Narrowband Vocoder Algorithm in an E3A Cabin Acoustic Noise Environment," RADC In-House Report No. RADC-TR-80-225, June 1980.

3.10 C.F. Teacher and H.Watkins, "ANDVT Microphone and Audio System Study," KETRON, Inc., Wayne, PA, August 1978; Final Report.

3.11 "USAF Bioenvironmental Noise Data Handbook," Vol. 127, "F-15A In-Flight Crew Noise," AMRL-TR-75-50, August 1979.

3.12 G.S. Kang and S.S. Everett, "Improvement of the Narrowband Linear Predictive Coder, Part 1—Analysis Improvements," NRL Report 8645, Naval Research Laboratory, Washington, 27 December 1982.

3.13 V.R. Viswanathan, K.F. Karnofsky, K.N. Stevens, and M.N. Alakel, "Multisensor Speech Input for Enhanced Immunity to Acoustic Background Noise," Proc. of 1984 IEEE ICASSP, IEEE Press, New York, 1984.

3.14 B. Widrow et al., "Adaptive Noise Cancelling: Principles and Applications," IEEE Proceedings, vol. 63, pp. 1692-1716 (December 1975).

3.15 R.C. Rohlfs, "Speech Enhancement Using the Widrow-Hoff Adaptive Noise-Cancelling Tapped Delay-Line Filter, a CSP-30 Implementation," MITRE Technical Report MTR-3626, April 1979.

3.16 R.J. McAulay and M.L. Malpass, "Speech Enhancement Using a Soft-Decision Noise Suppression Filter," IEEE Trans. ASSP, vol. 28, pp. 137-145 (April 1980).

3.17 T.E. Eger, J.C. Su, and L.W. Varner, "A Nonlinear Spectrum Processing Technique for Speech Enhancement," Proc. of 1984 IEEE ICASSP, IEEE Press, New York, 1984.

3.18 J.E. Porter and S.F. Boll, "Optimal Estimators for Spectral Restoration of Noisy Speech," Proc. of 1984 IEEE ICASSP, IEEE Press, New York, 1984.

* 3.19 G.F. Sandy and J.E. Parker, "Digital Voice Processor Consortium Interim Report," MITRE Corporation MTR-81W00159-01, McLean, VA, August 1982.

4.1 PDP 11/44 System User's Guide

* 4.2 J.D. Tardelli, C.M. Walter, J.W. Kennedy and J.T. Sims, "Research and Development for Digital Voice Processing," R84-01W ARCON Corporation, 31 July 1984; Final Report 1 January 1982 - 8 May 1984; and RADC-TR-85-46, March 1985.

5.1 G.S. Kang and S.S. Everett, "Improvement of the Narrowband Linear Predictive Coder, Part 1—Analysis Improvements," NRL Report 8645, Naval Research Laboratory, Washington, 27 December 1982.

5.2 G. Kang and S. Everett, "Improvement of the Narrowband Linear Predictive Coder: Synthesis Improvements", NRL Report 8799, June 1984.

5.3 D. Kemp, et. al., "Enhanced LPC-10 Voice Coder: Phase I Report", NSA Document R163-84-01, November 1984.

5.4 J. Wolf, K. Field, W. Russell, "Real-Time Implementation of the APC/SQ and LPC-10 Speech Coding Algorithms", BBN Report No. 4855, June 1982.

6.1 "General Information: SD350 Digital Signal Analyzer," Spectral Dynamics Corp., San Diego, CA, no date. Spectral Dynamics "SD350 Digital Signal Analyzer", March 1978.

6.2 Spectral Dynamics "Instruction Manual: GPIB Adapter for SD-350-6", April, 1980.

6.3 Digital Equip. Corp. "IBV11-A: LSI-11/ Instrument Bus Interface User's Manual", June 1977.

6.4 Adams-Russell Digital Processing Division, "Speech Processing Peripheral (SPP) User's Manual", Oct. 1984.

6.5 Signal Technology, Inc., "Interactive Laboratory System: User's Guide", March 1978.

6.6 Signal Technology, Inc., "Interactive Laboratory System: Programming Guide", October 1980.

* 6.7 J.D. Tardelli, C.M. Walter, J.W. Kennedy and J.T. Sims, "Research and Development for Digital Voice Processing," R84-01W ARCON Corporation, 31 July 1984; Final Report 1 January 1982 - 8 May 1984; and RADC-TR-85-46, March 1985.

6.8 Precision Filters, Inc., "Technical Manual 636-C-02 for Prgrammable Filters", October,1980.

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control, Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of competence is provided to ESD Program Offices (POs) and other ESD elements to perform effective acquisition of C^3I systems. The areas of technical competence include communications, command and control, battle management, information processing, surveillance sensors, intelligence data collection and handling, solid state sciences, electromagnetics, and propagation, and electronic, maintainability, and compatibility.

END
3-87
DTIC